NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

DISSERTATION

THE ADAPTIVE CONTROL OF OPTICAL BEAM JITTER

by

R. Joseph Watkins

December 2004

Dissertation Supervisor: Brij N. Agrawal

Approved for public release, distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** December 2004	**3. REPORT TYPE AND DATES COVERED** Dissertation	
4. TITLE AND SUBTITLE: The Adaptive Control of Optical Beam Jitter			**5. FUNDING NUMBERS**
6. AUTHOR(S) R. J. Watkins			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release, distribution is unlimited			**12b. DISTRIBUTION CODE A**

ABSTRACT *(maximum 200 words)* Acquisition, Tracking and Pointing (ATP) of space and airborne optical beams is becoming an important research topic as the requirements for the control of the beam is increasing. Arc-second accuracy and nano-radian jitter coupled with stringent pointing requirements test the limits of control systems. A unique Laser Jitter Control Testbed was built in order to study means to correct disturbances in a laser relay system. A new technique, the Adaptive Bias Filter (ABF), applies a compensating bias to the reference signal for the adaptive filter, allowing rapid convergence in correcting narrowband disturbances. A second technique, the Adaptive Delay Filter (ADF), was applied in a new way to the adaptive control systems to better correlate the error signal with the reference signal. Experiments using these techniques effectively demonstrated the usefulness of the new control methods, including the use of a combination Linear Quadratic Regulator (LQR) and LMS/ABF filter to remove disturbances caused by colored noise injected into an optical beam. The combination controller results in a 38 dB decrease in jitter caused by a 50 Hz vibration and a 10 dB decrease in an 87 Hz vibration of the supporting platform, and an overall 75% improvement in the broadband jitter experienced by the optical beam.

14. SUBJECT TERMS Adaptive Control, Optical Beam, Jitter, Least Mean Squares, Gradient Adaptive Lattice			**15. NUMBER OF PAGES** 188
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release, distribution is unlimited

THE ADAPTIVE CONTROL OF OPTICAL BEAM JITTER

R. Joseph. Watkins
Commander, United States Navy
B.S.ChE., Auburn University, 1983
M.S., Naval Postgraduate School, 1991

Submitted in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 2004

Author: _____
R. Joseph. Watkins

Approved by:

_____
Brij N. Agrawal
Distinguished Professor of
Mechanical and Astronautical
Engineering
Dissertation Advisor

_____
Young S. Shin
Professor of Mechanical and
Astronautical Engineering
Dissertation Co-Advisor

_____
Anthony J. Healey
Distinguished Professor of
Mechanical and Astronautical
Engineering

_____
Joshua H. Gordis
Associate Professor of
Mechanical and Astronautical
Engineering

_____
Fotis A. Papoulias
Associate Professor of
Mechanical and Astronautical
Engineering

Approved by: _____
Anthony J. Healy, Chairman, Department of Mechanical and
Astronautical Engineering

Approved by: _____
Juli Filizetti, Associate Provost for Academic Affairs

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Acquisition, Tracking and Pointing (ATP) of space and airborne optical beams is fast becoming an important research topic as the requirements for the pointing and control of the beam is increasing. Arc-second accuracy, nano-radian jitter and large flexible structures require stringent pointing requirements testing the limits of control systems. The equipment on board the "bus" results in narrowband as well as random structural interactions that further complicate the control method. The effect of the atmosphere on the beam adds a broadband disturbance, resulting in an optical beam that has been corrupted by "colored noise". A unique Laser Jitter Control Testbed was designed and built in order to study means to correct disturbances in a laser relay system. Several methods are implemented on the testbed, from classical control to adaptive systems, in order to develop and improve optical beam jitter control. In Adaptive control systems, the Least Mean Squares (LMS) and Gradient Adaptive Lattice (GAL) method have been used for disturbance rejection in electronic signals for years. However, these electronic signals are zero-mean and thus the effect of bias or phase difference in the error has been rarely studied. Without a compensator, a slowly varying bias in the error will result in a slow convergence to the steady state values. A phase difference between the reference signal and the error will prevent the GAL filter from removing energy in that tonal. A new technique called the Adaptive Bias Filter (ABF) applies a compensating bias to the reference signal for the adaptive filter, allowing rapid convergence in correcting narrowband disturbances. This technique works for both the LMS and the GAL filter, which were modified for use on the testbed. A second technique, called the Adaptive Delay Filter (ADF), was applied in a new way to the adaptive control systems to better correlate the error signal with the reference signal, resulting in faster convergence for low adaptation rates. Experiments using these techniques effectively demonstrated the usefulness of the new control methods, including the use of a combination Linear Quadratic Regulator (LQR) and LMS/ABF filter to remove disturbances caused by colored noise injected into an optical beam. The combination controller results in a 38 dB decrease in jitter caused by a 50 Hz vibration and a 10 dB decrease in an 87 Hz vibration of the supporting platform, as well as an overall 75% improvement in the broadband jitter experienced by the optical beam.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I must first thank God for the health and the opportunity to work on this dissertation. I consider myself extremely fortunate to have had the chance to complete a PhD while working for the Navy. Second, I wish to thank my wife Teresa, son Joseph and daughter Sarah for their constant support and the firm anchor they provided, especially when things didn't go as planned. This dissertation is dedicated to them.

I would like to express my most sincere gratitude to all the members of my Doctoral Committee. I must thank my advisor, Prof. Brij Agrawal and co-advisor, Prof. Young Shin, for their consistent encouragement which was so critical to completing this task. Prof. Anthony Healey was instrumental in understanding and implementing the "classical" control methods used in this research, as well as helping in understanding the basics of adaptive control. Prof. Joshua Gordis' and Prof. Fotis Papoulias' help in understanding control of flexible and vibrating structures was key in establishing baseline research for the new testbed. The author would also like to acknowledge the vital advice and help from Prof. Roberto Cristi in understanding and applying adaptive filters to the control problem. None of this research would have been possible without each of these men's contributions. Also providing valuable assistance in the optical engineering required for this research was Dr. Ty Martinez and Mr. Jeff Baker. Dr Eric Anderson and Mr. Mike Evert at CSA Engineering were instrumental in integrating the inertial actuator into the testbed. Mr. Jim Dillow at Boeing SVS provided invaluable insight into the practical aspects of laser beam control, and the demonstration provided by Boeing SVS at their New Mexico site was critical to designing and building the Laser Jitter Control testbed here at NPS. I would also like to thank Prof. Sen Kuo of Northern Illinois University for coming to Monterey to give us an important look at the computer control circuitry and some training on the use of adaptive filters. Finally, I would like to thank Dr. Hong-Jen Chen for his help, advice and trouble-shooting efforts during the beginning stages of the design and construction of this testbed.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. MOTIVATION

Many future space missions, laser communication systems, and imaging systems will require optical beam pointing and jitter control in the sub-micro-radian regime. Jitter is the undesired fluctuations in the pointing of an optical beam due to the environment or structural interactions, and consists of both broadband and narrowband disturbances. The control of jitter is also a challenging problem for current programs such as the Airborne Laser and the James Webb Space Telescope. Jitter has a great affect on the resolution of an image or the intensity of an optical beam. For example, a 100 mm diameter laser beam with 10 μ rad of jitter will result in roughly a 400 fold decrease in the intensity of the beam at 100 km, due to the jitter alone. In order to study this problem, the Naval Post Graduate School (NPS) developed the Optical Relay Mirror Lab – Laser Jitter Control (LJC) Testbed, located in the Spacecraft Research and Design Center (SRDC). Several methods are implemented on this testbed, from classical control to adaptive systems, in order to develop and improve optical beam jitter control. Proven adaptive means are tested and compared with the classical control model on disturbances provided by a randomly moving laser beam and vibration of the platform containing the control system. The control system consists of Position Sensing Detectors (PSD) and a Fast Steering Mirror (FSM) which are used to correct the beam's deviations from the center of a target PSD. Several techniques are developed to improve the performance of the adaptive systems.

At the NPS SRDC there is a major research program on optical beam pointing and jitter control. The current application is the Bifocal Relay Mirror spacecraft. This spacecraft is composed of two optically coupled telescopes used to redirect the laser light from ground-based, aircraft-based or spacecraft-based lasers to distant points on the earth or in space for a variety of non-weapon force enhancement missions. Based on an Air Force Research Laboratory (AFRL) study, the potential missions are camouflage detection and penetration, chemical warfare agent detection and identification,

illuminator for night time/active imaging, laser fence for aircraft detection, and tunnel and underground structure detection. Students at NPS developed a design for the Bifocal Relay Mirror Spacecraft in the summer of 2000 with AFRL sponsorship. The effort identified several unique technologies to be developed on acquisition, tracking and pointing as well as beam control optics. The SRDC submitted a proposal to the National Reconnaissance Office to develop these unique technologies for a Bifocal Relay Mirror Spacecraft (BRMS) with AFRL as the co-investigator for beam control optics. The NPS/AFRL proposal was selected for the award in December 2000. The SRDC designed and had built a BRMS simulator to validate experimentally the control methods necessary to accurately stabilize the platform and meet the stringent optical beam pointing requirements.

The SRDC has conducted research in the past on active control of narrowband disturbances in structures. However, the optical beam in a relay system will have broadband noise in addition to the narrowband jitter that results from rotating equipment on the spacecraft bus or aircraft platform. The object of this research is to develop improved control techniques to remove both the narrowband and broadband jitter in the optical beam. These techniques will be validated both analytically and experimentally.

B. LITERATURE REVIEW

1. Background

The control of disturbances from unwanted vibration has its beginning in the area of acoustics. The first methods used to control vibrations were in an attempt to reduce unwanted noise (acoustic vibration). These methods began with the use of passive systems, basically blanketing the area to be protected with material that would absorb the frequencies of interest. Using a damping material to reduce structural vibrations in a similar manner is a useful technique for structural disturbances. These types of systems work well if size and weight are not an issue, but this is not the case for airborne or space borne systems.

The earliest attempts to actively control sound are traced to Paul Lueg of Germany who tried to cancel sound propagating through a duct using a 180 degree phase-

shifted sound wave to destructively interfere with the source wave [Lueg, 1937]. Various other attempts have been made using similar techniques, with those involving multiple cancellation sources enjoying the best results.

The availability of faster and better processors resulted in the use of digital signal processing techniques to actively control sound, and for the first time in the 1970s, structural vibration. By the 1980s, there were several designs of active vibration control for systems such as the classical spring-mass damper system [Burdess, 1985] vibration isolation platforms [Takagami, 1985], flexible beams [Schafer, 1985], and others. The new techniques depended heavily on the ability to model the physical structure, sensors and actuators involved. This approach suffered from the inability to accurately model the system, or from changes in the system or environment after the model was determined. Many physical systems are too complex to model accurately using analytical methods, leading to simplifications that could severely impact the effectiveness of the control system. One possible answer to this problem was to use an adaptive control system, one that tunes itself to the system and adapts itself to changes in the system or environment automatically.

Control of optical beams has only relatively recently been under consideration in the research community. As the amount of bandwidth necessary to transmit information has increased exponentially over the last decade, the use of laser beams for communication has become increasingly attractive. Directed energy has always been the plaything of science fiction, but now with the advent of new powerful lasers, the fantasy could become reality. However, unlike a radio wave, which spreads in a spherical pattern, or a missile, which can make in-flight corrections, the laser beam only goes in the direction you point it. Even if you can point it accurately enough, you have to deal with the vibrations that are inherent in mechanical systems. Jitter, as was stated earlier, reduces the intensity of the beam at the target, whether the beam is used as a communications system or a weapon.

2. Classical Control

A straight-forward method of controlling jitter is to use the classical means of control such as Proportional-Integral-Derivative (PID) control or a Linear-Quadratic-Regulator (LQR). The PID controller works well for low frequency (less than 10 Hz) disturbances on the LJC, but becomes less capable at the higher frequencies. An LQR controller works well for controlling even at high frequencies (less than about 200 Hz on the LJC), but requires full state feedback (not normally available) or a means to estimate the states not observed [Ogata, 2002, pp. 897-910]. The LQR controller requires an accurate model in order to solve for the gains necessary to control the system. The ability to effectively control the system depends on the analytical or numerical model and the estimation filter. As these may be inaccurate at frequencies not allowed for, or changes in the environment that are not expected, the LQR may require "tuning" in order to maintain the desired accuracy. This tuning would not be desired in a weapons system or a space borne autonomous platform.

McEver and Clark used a Linear Quadratic Gaussian (LQG) design to actively control jitter in an experiment designed around Fast Steering Mirrors, accelerometers and microphones. The disturbance was acoustically induced by loud speakers and control was attempted using feedforward from a microphone or the accelerometers to the FSM as well as feedback to the FSM using a Position Sensing Detector (PSD) at the target. Acoustic means of controlling the jitter were also attempted using a secondary loudspeaker and feedback from a microphone located near the control mirror. The FSM employed by McEver used a piezo-electric actuator, unlike the LJC FSMs which use voice coils to move the mirror. McEver's experiment showed that using feedforward FSM control with the microphone resulted in a reduction from 4.4 to 2.5 rms μrad broadband for the y-axis of the PSD. He did not specify any reduction in the x-axis. Feedback FSM control using the PSD reduced the x-axis jitter from 2.51 to 1.38 rms μrad and the y-axis jitter from 7.0 to 2.7 rms μrad. The use of Acoustic means to control the jitter did not result in any reduction [McEver, 2001].

Glaise, Anderson and Janzen conducted similar experiments as McEver, et al using acoustically induced jitter and LQG control. Their FSM also used piezo-electric actuators. Feedback control was attempted using a secondary loudspeaker as an actuator as well as the FSM. The feedback was provided by a PSD. The use of a secondary loudspeaker did not result in appreciable improvement of the jitter recorded by the PSD. With the FSM as the actuator, the reduction in jitter was considered good, with an almost complete elimination of a 30 Hz response on the PSD. The LQG controller had been targeted for this frequency based on the open loop response observed as a result of the acoustic disturbance [Glaese, 2000].

3. Adaptive Control

In the early 1950s there was extensive research into methods of controlling high-performance aircraft using adaptive means. These aircraft operated over a wide range of altitude and speed. In one regime, a linear, constant-gain classical feedback system might work, but it would not work over the complete range of flight parameters. A more sophisticated controller was needed and a system of "gain scheduling" was devised that worked quite well. The controller thus "adapted" to its environment, allowing one system to be used to control a widely varying flight regime. This was a very rudimentary form of adaptation, but due to the limitations of computer processors at that time, was the best that could be done. The introduction of State-Space methods, Stability Theory and Stochastic Control Theory were major developments in the 1960s that aided resurgence into adaptive control research in the 1970s. In the late 1970s and early 1980s, proofs for stability of adaptive systems appeared. As processor speed and memory storage capability increased, the opportunity to use adaptive means improved. As a result, adaptive controllers started to appear commercially in the early 1980s [Astrom, 1995].

Adaptive control has also been used in the control of optical beam jitter. Skormin, Tascillo, and Busch developed a computer simulation in 1995 of an airborne to satellite optical link in which the use of a self-tuning regulator (STR) as well as a Filtered-X Least Means Squares (FXLMS) controller was used to mitigate the effects of jitter on the optical beam. The simulation shows that adaptive feedforward vibration compensation can be used to minimize induced jitter [Skormin, 1995]. In 1997, Skormin

and Busch proposed the use of Model Reference Control for jitter reduction. Experiments were conducted using a specially designed high bandwidth FSM and a commercially available low bandwidth FSM. In each case, significant reduction (on the order of 20 dB) in acoustically induced jitter was achieved [Skormin, 1997a]. Also in 1997, Skormin, Tascillo and Busch demonstrated the use of a self-tuning regulator in an acoustically induced jitter rejection experiment using FSMs (voice-coil actuated), accelerometers and PSDs. The data showed that the STR was superior to classical feedback control in frequency ranges above about 300 Hz. [Skormin, 1997b].

a. Least-Mean-Square (LMS) Method

Widrow and Hoff proposed the LMS algorithm in 1960 [Widrow, 1960]. The LMS method uses a Finite Impulse Response (FIR) filter to generate a control signal from a reference input. The FIR gains are adjusted by the algorithm to minimize the Mean Square Error (MSE) of the output. The algorithm is elegant in its simplicity, requiring neither measurements of the pertinent correlation functions, or matrix inversion. The LMS algorithm was originally designed for signal processing, and is the standard against which other linear adaptive methods are measured [Haykin , 2002, p. 231]. When used as an adaptive noise or vibration canceller, the effects of the secondary path (the actuator) must be taken into account to improve performance. This led to the Filtered-X version (FXLMS) of the algorithm in which the reference signal is filtered by a model of the secondary path. Widrow et al derived the algorithm in 1981 [Widrow, 1981] in the context of adaptive control and Burgess independently derived it for use in noise control [Burgess, 1981]. The LMS algorithm uses a reference signal for removal of the unwanted disturbance. This reference signal has been the subject of many papers. One example that is applicable to the rejection of colored noise in a laser beam is the use of amplitude optimization to speed up convergence for the multiple frequency case of an ANC system [Kuo, 1998b]. . In this paper, Kuo et al, discuss the use of phase compensation to reduce out of band overshoot for a single frequency ANC system. As we shall show, the phase, amplitude and bias of the reference signal are critical to convergence in a targeting scenario.

The FXLMS algorithm has also been used in the control of optical beams. As mentioned earlier, Skormin, et al, used it in the simulation of an airborne platform to satellite link. Glaese, et al, used it in the experiment referenced earlier in which LQG control was used. In Glaese's experiment, FXLMS was implemented with the FSM as an actuator. The performance varied, depending on the reference source used, from a 32 to 47 dB reduction in jitter of an 80 Hz tone induced by the loudspeaker. The broadband reduction varied from 7.4 to 9.5 dB [Glaese, 2000]. Boelitz, et al conducted airborne experiments using the FXLMS algorithm to reject disturbances from the vibration of the airframe and mounted optical bench during actual flight conditions. These experiments resulted in varying degrees of performance, from a 25% to 99% reduction in the Normalized Correlated Power of the track error. In some cases the jitter actually increased. However, due to the complexity of the experiment, data is still being analyzed and the increase in jitter could be due to a loss of fine track mode during the data run [Boelitz, 2003]

b. Adaptive Lattices

In 1978, Griffiths, (1978) and in a separate paper, Makhoul, (1978) proposed the use of a Lattice Filter to solve a joint-process estimation problem that is optimal in the mean-square error sense. The Lattice Filter develops a vector of backward prediction errors which are orthogonal to each other, simplifying the solution to the problem significantly, resulting in parameters that converge faster than the normal linear transversal filter used in the LMS algorithm [Haykin, 2002, p. 543]. The filter developed by Griffiths is a member of the stochastic-gradient family of adaptive filters, of which the LMS filter is a part. These lattice filters are, however, computationally extensive, and in some cases, (specifically filters of higher orders), prohibitively so [Watkins, 2004]. Jiang and Gibson developed a multi-channel, unwindowed lattice filter in which the recursive least-squares (RLS) estimation method is used. The RLS method is more exact than the stochastic gradient method, but requires the inversion of the error-covariance matrix. Jiang and Gibson's lattice filter uses a modified Gram-Schmidt procedure to eliminate the necessity of inverting any matrices [Jiang, 1995]. Boelitz, et al used Jiang and Gibson's lattice filter in an airborne experiment called ABLE ACE to control structural vibrations

on an optical train. The results show a reduction in the Normalized Correlated Power of the track error from 16% to 95% [Boelitz, 2003].

C. THESIS OVERVIEW

Chapter II presents an outline of the basic theory used in the control algorithms that are implemented in the course of this research. Since the Linear Quadratic Regulator is widely known and used, only the basics of this algorithm are discussed. The LMS filter and its application to control theory will be reviewed, as well as the development of the Filtered-X modification. Additionally, the effects of a phase-shift of the reference signal are explored, as well as the effects of changing the amplitude of the reference signal. The Gradient Adaptive Lattice is then derived and its implementation considered.

Chapter III will discuss the experimental setup of the Laser Jitter Control testbed. The sensors and actuators, as well as the computer control system and support structure will be explained. Chapter IV details the system identification experiments and results. A state-space model of the system will be derived, and the system used to calibrate the LJC will be detailed.

In Chapter V, an analogous model will be derived using a simple mass-spring-damper system. This model will be used for investigations into the characteristics of the different control schemes. The analytical solution to the model will be presented and numerical simulations analyzed to determine the effectiveness of the different classical and adaptive control methods.

Chapter VI will discuss the disturbance rejection experiments attempted on the LJC. The experiments are categorized into three groups; those involving an external disturbance to the system, those in which the platform containing the control system is disturbed, and those in which both external and platform disturbances occur. Chapter VII will provide conclusions, and recommendation for future work.

II. REVIEW OF THEORY

A. LINEAR QUADRATIC REGULATORS

The LQR is first developed to investigate how classical control algorithms handle broadband and narrowband disturbances. The system to be controlled must be modeled mathematically to determine the optimal gains. The LQR requires full state feedback, which, if not available, must be estimated. A Kalman filter or other observer may be used to estimate the states not measured. The linear-quadratic optimal gains (K) are calculated to minimize the following cost function:

$$J = \int_0^\infty \left(\bar{x}^T Q \bar{x} + \bar{u}^T R \bar{u} \right) dt \tag{2.1}$$

The control law is:

$$\bar{u} = -K\bar{x} \tag{2.2}$$

In order to obtain the optimum gains, a state-space model of the system must be determined. The state-space model is of the form:

$$\begin{aligned} \dot{\bar{x}} &= A\bar{x} + B\bar{u} \\ \bar{y} &= C\bar{x} + D\bar{u} \end{aligned} \tag{2.3}$$

where ***x*** is the state vector and the matrices ***A*** and ***B*** describe the system. ***y*** is the output of the system, and the matrices ***C*** and ***D*** describe the effect of the states and the input on the output. In equation (2.1), the matrices ***R*** and ***Q*** are weighting matrices, to provide information on the relative importance of each state and the input to the control effort. Therefore, in order to optimally determine the gains for a specific response, one must understand the relationship between the state vector and the physical quantities involved. Accordingly, the following model was developed for the LJC.

A simple second order transfer function of the Newport mirror about one axis is defined:

$$H_m(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{2.4}$$

where ω_n and ζ are experimentally determined. A first order system was added to model the optical sensor system:

$$H_d(s) = \frac{1}{Ts + 1} \tag{2.5}$$

where ***T*** is the time constant for the PSD system. The value for ***T*** was determined from the data given for the optical sensors. The resulting State-Space set of equations for one axis is given:

$$\begin{bmatrix} \dot{V}_p \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} -1/T & 2GpDm/T & 0 \\ 0 & 0 & 1 \\ 0 & -\omega_n^{\,2} & -2\zeta\omega_n \end{bmatrix} \begin{bmatrix} V_p \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega_n^{\,2}Gm \end{bmatrix} V_m + \begin{bmatrix} -GpDb/T \\ 0 \\ 0 \end{bmatrix} \theta_d \tag{2.6}$$

In equation (2.6), V_p is the voltage at the optical sensor, ***Gp*** is the optical sensor gain, V/m, and D_m is the distance from the control mirror to the sensor, m. θ is the rotation about the control mirror's x axis, ***Gm*** is the control mirror gain, rad/V, and V_m is the voltage input to the control mirror. D_b is the distance from the disturbance source to the detector and θ_d is the rotation about the laser beam's x or y axis at the disturbance source. θ_d is a pseudo-random variable, considered to be band-limited white noise plus any narrow-band disturbances. The state-space system defined above is used to solve for the optimal gains for the LQR, with increased weighting in the matrix **Q** in equation (2.1) on the voltage at the detector, V_p.

B. THE LEAST MEANS SQUARES (LMS) ALGORITHM

The LMS algorithm is one of the simplest yet robust adaptive algorithms. In the LMS algorithm, we adjust the *tap gains* based on the response of the system to (1) the error, (2) a reference signal correlated with the disturbance, and (3) the control input. The algorithm uses the method of Least Squares to find the optimum values for the tap gains. In particular, the algorithm relies on *predicting* its next input, which is simply the disturbance in the case of laser jitter control, to optimize the *tap gains*. The error - the

difference between the predicted signal and the system output - is then used to recalculate the gains that minimize the error in return. For the LJC Testbed, the feed-back or target detector is used to provide the error signal, and the feed-forward detector provides the correlated disturbance input signal. This type of control algorithm not only calculates the necessary gains, but also identifies the system, simplifying the requirement to mathematically model the system. The type of adaptive control employed in the experimental setup utilizes predictors. One type of *forward* predictor is the transversal or ladder filter, as shown in Figure 1. One may also conduct *backward prediction*, i.e. calculate what an input was in the past, given the **n** current and past inputs.

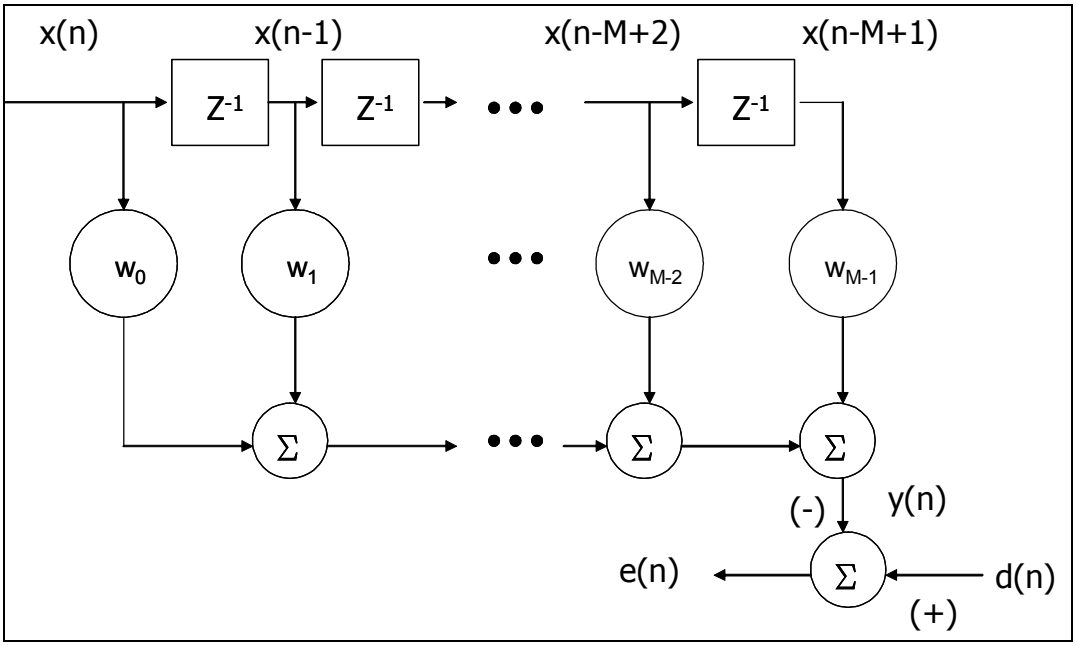


Figure 1 Transversal Filter

The derivation that follows is modeled after Kuo (1996), modified for the optical jitter control system. Define an input vector at time n as:

$$\bar{x}(n) \equiv \left[x(n) \quad x(n-1) \cdots \quad x(n-M+1) \right]^T \tag{2.7}$$

where M is the order of the filter, (number of weights), and the superscript T denotes transpose. Furthermore, the over-bar designates a vector. Define the weight vector at time n as:

$$\bar{w}(n) \equiv \left[w_0(n) \quad w_1(n) \cdots \quad w_{M-1}(n) \right]^T \tag{2.8}$$

The output *y(n)* of the transversal filter in Figure 1 is then:

$$y(n) = \bar{w}^T(n)\bar{x}(n) \tag{2.9}$$

This output is compared with a desired signal, *d(n)* which results in the error equation:

$$\begin{aligned} e(n) &= d(n) - y(n) \\ &= d(n) - \bar{w}^T(n)\bar{x}(n) \end{aligned} \tag{2.10}$$

Our objective will be to minimize the Mean Square Error, $\xi(n)$ by adjusting the tap gains, $\bar{w}(n)$. We will restrict the statistics of $\bar{x}(n)$ and *d(n)* to be stationary and the vector $\bar{w}(n)$ to be a deterministic sequence. Define the Mean Square Error, (MSE) by:

$$\xi(n) = E\left[e^2(n)\right] \tag{2.11}$$

where *E*[ ] denotes the expected value. Substituting equation (2.10) into (2.11)gives:

$$\xi(n) = E\left[d^2(n)\right] - 2E\left[d(n)\bar{x}(n)\right]^T \bar{w}(n) + \bar{w}^T(n) E\left[\bar{x}(n)\bar{x}(n)^T\right]\bar{w}(n) \tag{2.12}$$

Now define the desired-to-input cross-correlation vector and the input autocorrelation matrix as:

$$\bar{p} \equiv E\left[d(n)\bar{x}(n)\right] \tag{2.13}$$

$$\mathbf{R} \equiv E\left[\bar{x}(n)\bar{x}(n)^T\right] \tag{2.14}$$

where the boldface letter indicates a matrix. Equation (2.12) may now be written as:

$$\xi(n) = E\left[d(n)^2\right] - 2\bar{p}^T\bar{w}(n) + \bar{w}(n)^T \mathbf{R}\bar{w}(n) \tag{2.15}$$

We may now determine the optimum tap gains that minimize the MSE by performing a vector differentiation of $\xi(n)$. This results in the *Wiener-Hopf* equations, formulated for the discrete-time filter:

$$\overline{w}^{o} = \mathbf{R}^{-1}\overline{p} \tag{2.16}$$

the superscript *o* denoting the optimum values for the weight vector. In order to obtain the optimum, we must have the values for **R** and $\overline{p}$, which in practice, in real time, are difficult if not impossible to evaluate accurately. We must have a means to estimate **R** and $\overline{p}$.

It can be shown that the MSE is a quadratic function in the weight vector, and that any departure from $\overline{w}^{o}$ will result in an increase in the MSE. The MSE surface has only one global minimum point, and at that point, the tangents to the surface must be zero. We may therefore use gradient-based algorithms to search out the optimum weight vector [Kuo, 1996, pp. 22-23]. The LMS algorithm makes use of the *steepest descent* method to find the optimum weight vector. The steepest descent method takes "steps" in the direction of the negative of the gradient of the surface, here the MSE surface. We update the weight vector as follows:

$$\overline{w}(n+1) = \overline{w}(n) - \frac{\mu}{2}\nabla\xi(n) \tag{2.17}$$

where μ is the step size that controls stability of the filter as well as the rate of descent along $\nabla\xi(n)$. The factor 2 is used to simplify the final result. From equation(2.15) we may obtain $\nabla\xi(n)$:

$$\nabla\xi(n) = -2\overline{p} + 2\mathbf{R}\overline{w}(n) \tag{2.18}$$

However, as stated earlier, we need to find a means to estimate $\overline{p}$ and **R**. Widrow (1970) used the instantaneous squared error as the estimate for the MSE:

$$\hat{\xi}(n) = e^{2}(n) \tag{2.19}$$

The estimate of the gradient of the MSE is thus:

$$\nabla\hat{\xi}(n) = 2\left[\nabla e(n)\right]e(n) \tag{2.20}$$

and, noting that $\nabla e(n) = -\bar{x}(n)$ (see equation (2.10)), we find that the update equation for the weight vector to be:

$$\bar{w}(n+1) = \bar{w}(n) + \mu \bar{x}(n) e(n) \tag{2.21}$$

which is known as the LMS algorithm. The block diagram of the algorithm in a form for implementation on the LJC is shown in Figure 2:

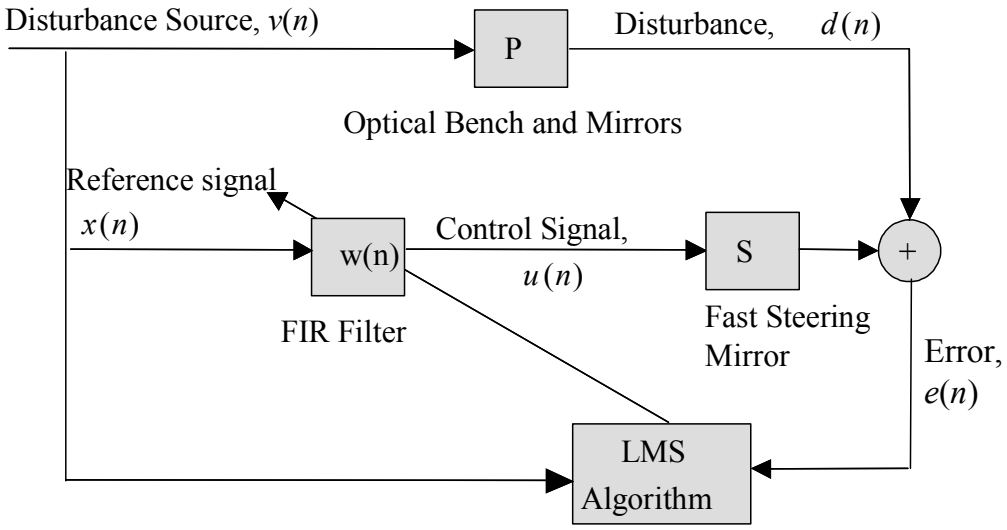


Figure 2 LMS block diagram

P denotes the *primary plant* transfer function which the disturbance must pass through before the output, and S denotes the *secondary plant* or actuator transfer function the control signal must pass through before the summing junction. P includes the transfer function of the structure of the bench and mirrors for the case of vibration of the bench, the transfer function of the sensors, as well as the gain effect caused by the distance the light beam travels through the system to the target. S must include the effect of vibration to the control mirror, the transfer function of the mirror and sensors, and the delays inherent in the digital signal processing and the computation of the control signal. In Figure 2 it may be seen that the output of the FIR filter does not go directly to the

summing junction for the error. Therefore the conventional LMS algorithm must be modified to ensure convergence. Taking the Z-transform of the signals in Figure 2, we find that the error signal is:

$$E(z)=\left[P(z)-S(z)W(z)\right]X(z) \tag{2.22}$$

Thus $W(z)$ must model the plant as:

$$W(z)=\frac{P(z)}{S(z)} \tag{2.23}$$

in order to drive $E(z)$ to zero. We note here that it is impossible to compensate for the delay caused by S if the primary plant P does not contain a delay of at least equal length. For the control of optical jitter, the delay in S can result in severe difficulties, due to the velocity of light, which is nearly instantaneous when compared with the speed of the Fast Steering Mirror (FSM) and control circuitry. And in general, if S is zero for some frequency, the controller will be unstable. Therefore, the effects of the secondary plant must be taken into account.

A simple solution to the problem is to provide an inverse filter ($1/S(z)$) in series with $S(z)$. However, the inverse does not always exist. A second solution would be to place a copy of the secondary plant transfer function in the path to the updating algorithm for the weight vector. This is known as the Filtered-X LMS (FXLMS) algorithm and was derived by Widrow (1981) and Burgess (1981). The block diagram is provided below:

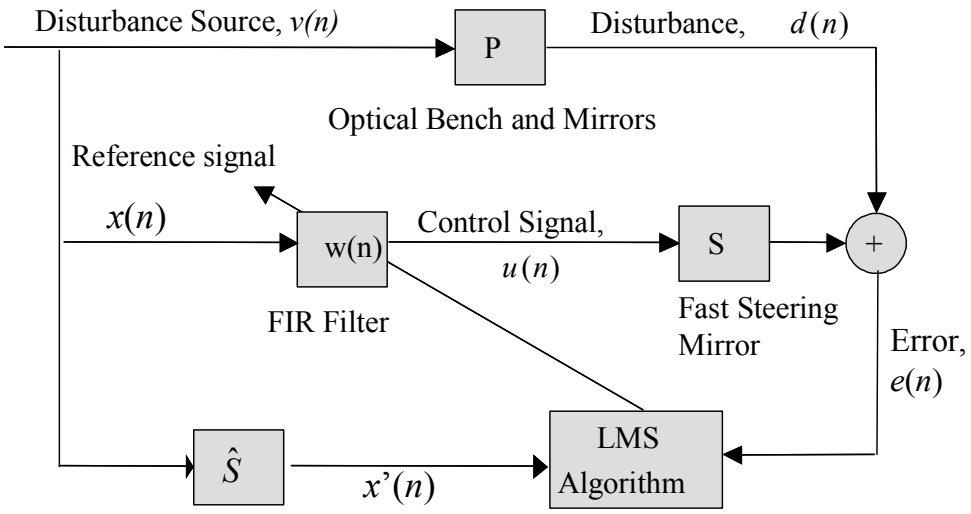


Figure 3 FXLMS Block Diagram

The update equation for the weight vector (equation(2.21)) now becomes:

$$\bar{w}(n+1) = \bar{w}(n) + \mu \bar{x}'(n) e(n) \tag{2.24}$$

with $\bar{x}'(n)$ defined as:

$$\bar{x}'(n) = \hat{\bar{s}}(n) * \bar{x}(n) \tag{2.25}$$

and $\hat{\bar{s}}$ being the estimate of the impulse response function of the secondary plant, the * denoting convolution.

The value of the step size μ has been the subject of no little study. μ effects both the rate of convergence as well as the stability of the algorithm. The correct selection of a step size is usually done with knowledge of the maximum value for stability, and then by trial and error. There have been a few studies done on the optimum value of μ for different cases, (see for example Haykin (2002), pp. 324-325) but these are in general hard to implement in real time. For the FXLMS algorithm, Elliott and Nelson (1993) have shown that the maximum step size is approximately:

$$\mu_{\max} = \frac{1}{E\left[\bar{x}'^2(n)\right](M+\Delta)} \tag{2.26}$$

where Δ is the delay in the secondary path, and M is the filter length. Another means for determining the step size is to use the normalized LMS algorithm. In this scheme, the step size is varied according to the input power of the reference (or filtered reference) signal. μ becomes a function of the iteration, n, and may be calculated as [Kuo, 1996, pp. 33-35]:

$$\mu(n) = \frac{\alpha}{\bar{x}^T(n)\bar{x}(n)} \tag{2.27}$$

where the parameter α is bounded by $0 < \alpha < 2$. The normalized FXLMS (nFXLMS) algorithm is used in most of this research.

Another factor in the selection of the step size is the *Misadjustment,* M_ξ. The misadjustment is the ratio of the excess MSE to the minimum MSE. The excess MSE is caused by the algorithms inability to converge to the optimum weight vector, $\bar{w}^o$. Since we are using the steepest descent method, there will be a certain amount of gradient noise due to the fact that the gradient of the MSE must be estimated. This noise results a MSE greater than the MSE that results from the Wiener-Hopf solution. M_ξ can be shown to be directly proportional to μ as:

$$M_\xi = \frac{\mu}{2}\sum_{k=1}^{M}\lambda_k \tag{2.28}$$

where λ_k are the eigenvalues of the autocorrelation matrix **R** and M is the filter length. As μ is increased, the misadjustment increases and the excess MSE becomes greater. A second factor in the selection of μ is the time constant, or rate of convergence. The average time constant (the time it takes for the MSE to decrease by a factor of 1/e from its starting value) is calculated as:

$$\tau_{mse} \approx \frac{1}{2\mu\lambda_{av}} \tag{2.29}$$

where λ_{av} is the average of the eigenvalues of **R** [Haykin, 2002, pp. 271-272]. From equations (2.28) and (2.29) we see that as μ is increased, the excess MSE increases, but the average time constant decreases. There is a tradeoff in the selection of the step size.

1. Effect of Phase Shifting the Reference Signal

Let us define an input vector as:

$$\bar{v}(n) = \begin{bmatrix} v(n) & v(n-1) & \cdots & v(n-M+1) \end{bmatrix}^T \tag{2.30}$$

This vector is the actual disturbance before the measurement that results in $\bar{x}(n)$. We restrict $\bar{v}(n)$ at this time to be the vector of discrete samples that result from a periodic, zero-mean disturbance. We now change our FXLMS block diagram to include an Internally Generated Reference Signal (IGRS) that generates $x(n)$. In other words, sensors detect $v(n)$ and generate $x(n)$. Now, as discussed earlier, we note that any error in the optical bench's position or the laser beams position is instantaneously (relative to the FSM's velocity) measured at the optical error sensor. This effect results in a violation of causality. However, as long as the beams position changes slowly enough, one can correct the error before it becomes too large. The block diagram of the system looks like this:

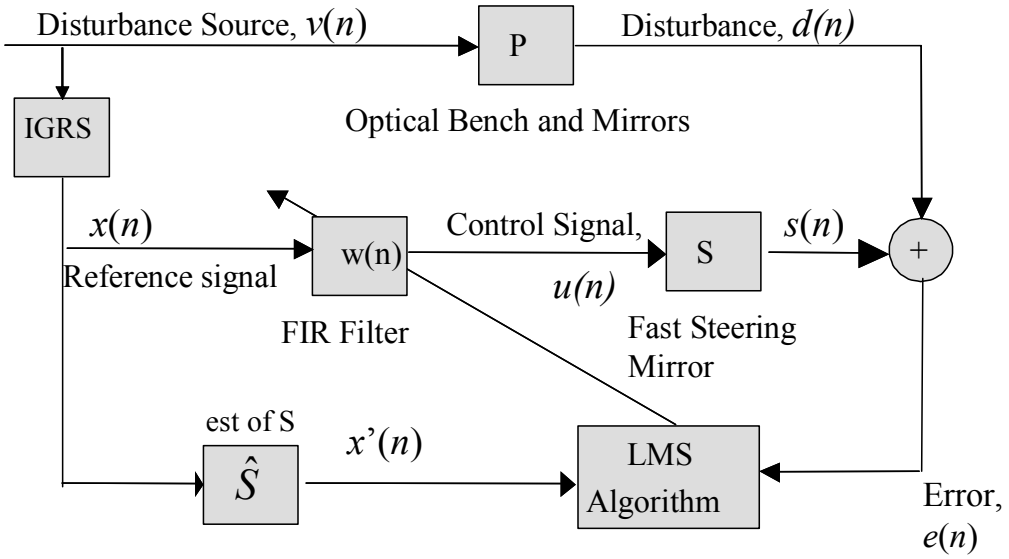


Figure 4 FXLMS with IGRS

Let $v(n)$ be $A\sin(2\pi fnT_s)$. Then, *d(n)* will be $PA\sin(2\pi fnT_s+\phi_p)$ where T_s is the sample time, *f* is the frequency in cycles/sec and ϕ_p is the phase delay through the primary plant. Now, let the generated reference signal, $x(n)$ be:

$$x(n)=B\sin(2\pi fnT_s+\phi_G) \tag{2.31}$$

where ϕ_G is to be determined. The output of the FIR filter generated by the LMS algorithm is:

$$u(n)=WB\sin(2\pi fnT_s+\phi_G+\phi_W) \tag{2.32}$$

where ϕ_W is the phase delay through the FIR filter. Let the output of the secondary plant (the FSM and DSP circuitry) be *s(n)*. With input *u(n)*, *s(n)* will be:

$$s(n)=SWB\sin(2\pi fnT_s+\phi_G+\phi_W+\phi_S) \tag{2.33}$$

where ϕ_S is the phase delay through the secondary plant. In order for e(n) to be zero, $s(n)=d(n)$ which results in:

$$SWB=PA \tag{2.34}$$

and:

$$\phi_G+\phi_W+\phi_S=\phi_p \tag{2.35}$$

The function of the LMS algorithm it to adapt *W* and ϕ_W in equations (2.34) and (2.35) to accomplish compete rejection of the disturbance signal, i.e.

$$W=\frac{PA}{SB} \tag{2.36}$$

and:

$$\phi_W=\phi_p-\phi_G-\phi_S \tag{2.37}$$

Therefore, if one could determine appropriate values for W and ϕ_W such that the algorithm started the recursion with tap gains near the optimum values, the time it would take for the algorithm to reach steady state would be reduced.

Now, returning to the weight vector update equation (equation (2.24)) we see that the rate of update of the vectors is dependent on three factors for the nFXLMS algorithm:1) the value of μ, 2) the input power of the reference signal, and 3) the correlation between $\bar{x}(n)$ and $e(n)$. We note that $\bar{x}'(n)e(n)$ in equation (2.24) is a rough estimate of the desired MSE gradient, however:

$$E\left(\hat{\nabla}\xi^2\right) = E\left(e(n)\bar{x}'(n)\right) = \mathbf{R}\bar{w}(n) - \bar{p} = \nabla\xi^2 \tag{2.38}$$

and it is therefore an *unbiased* estimate [Shan, 1988]. If we maintain μ and the reference signal power constant, then the adaptation rate is strictly dependent on the correlation between $\bar{x}'(n)$ and $e(n)$. When the correlation is the greatest, the adaptation rate of the weight vector is greatest, and when $\bar{x}'(n)$ is uncorrelated with $e(n)$, the adaptation of the weight vector stops. Since the change in the weight vector is always in the direction of the estimate of the negative gradient of the MSE surface, we conclude that by adjusting ϕ_G to better correlate $\bar{x}'(n)$ with $e(n)$ the rate of convergence to the optimum weight vector may be increased. This effect decreases with increasing μ, but the use of a large μ increases the misadjustment. We can therefore decrease the time constant of the process without increasing the misadjustment.

a. The Adaptive Delay Filter

The question now becomes what to set for a value of ϕ_G? To start with, let us evaluate the phase of the error:

$$\begin{aligned} e(n) &= d(n) - \bar{w}^T(n)\bar{x}'(n) \\ &= PA\sin\left(2\pi fnT_s + \phi_p\right) - SWB\sin\left(2\pi fnT_s + \phi_G + \phi_W + \phi_S\right) \end{aligned} \tag{2.39}$$

Through the use of trigonometric identities, one may determine the phase shift of the error (ϕ_e) to be:

$$\tan(\phi_e) = \frac{PA\sin(\phi_p) - SWB\sin(\phi_G + \phi_W + \phi_S)}{PA\cos(\phi_p) - SWB\cos(\phi_G + \phi_W + \phi_S)} \tag{2.40}$$

The evaluation of ϕ_W is difficult, since it is being adapted by the LMS algorithm, but at the outset of the adaptation process, it is easy to see that $\phi_W = \phi_G$. If we set the initial values of $\overline{w}(n)$ such that they are all equal (nominally zero), the $\overline{w}(n)$ will just be an attenuated $\overline{x}(n)$ after the first adaptation step, and therefore, $\phi_W = \phi_G$. The correct ϕ_G will be that which equals ϕ_e, resulting in the optimum correlation between $e(n)$ and $\overline{x}(n)$. We may thus plot the value of $\tan(\phi_G)$ verses the $\tan(\phi_e)$ as a function of ϕ_G to find the value that maximizes the first adaptation step, assuming we know ϕ_p and ϕ_S. However, at the next adaptation step, ϕ_W begins to change, and the correct value of ϕ_G is no longer known. We also may not know the precise value of ϕ_p and ϕ_S, or their values may change with time due to changes in input frequency, etc. ϕ_G should be adaptable in order to take advantage of this effect. Kuo and Chung [Kuo, 1998b] developed an Adaptive Delay Filter (ADF) that used one variable delay element and one variable gain element to estimate the time delay and gain difference between the primary and secondary paths. However, their filter used the time delay element and gain element to replace the FIR filter $W(z)$. We make use of this adaptive delay element in a different scheme to vary the generated reference signal phase as can be seen in the block diagram below:

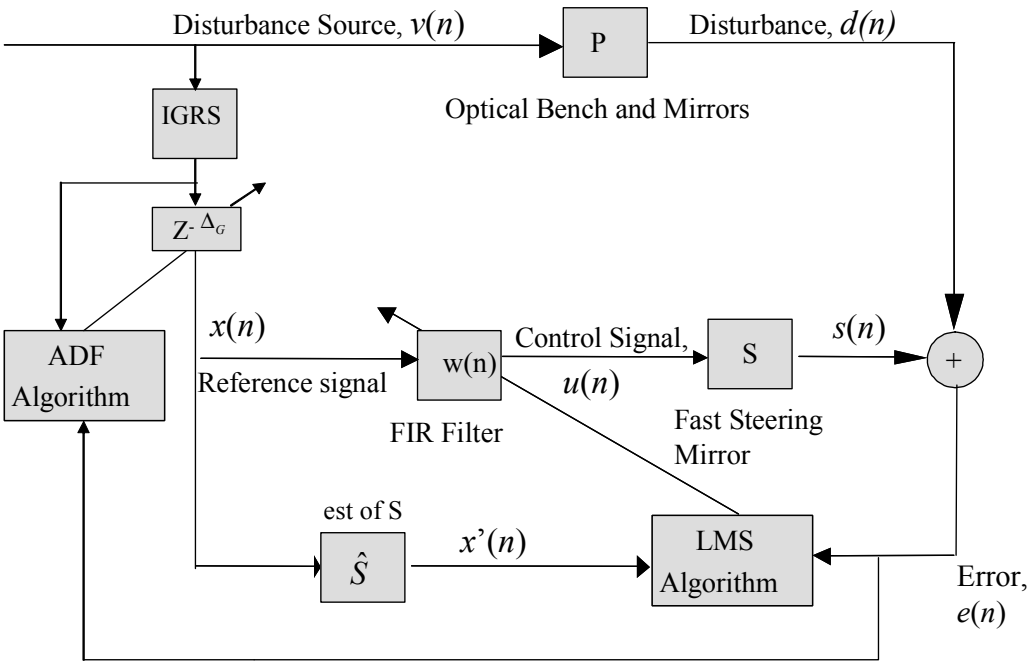


Figure 5 nFXLMS with ADF

In the weight vector update equation for the ADF, a symmetric difference is used to approximate the gradient to reduce the influence of bias from gradient estimation [Etter, 1981]. In the ADF algorithm, there is only one output, the delay that corresponds to ϕ_G, Δ_G. The update equation for Δ_G is:

$$\Delta_G(n+1) = \Delta_G(n) + \mu_\Delta e(n)\left[x\left(n-\Delta_G(n)-\Delta_S-1\right)-x\left(n-\Delta_G(n)-\Delta_S+1\right)\right] \quad (2.41)$$

where μ_Δ is the adaptation rate for the ADF , $e(n)$ is the error, and Δ_S is the delay through the FSM and control circuitry. μ_Δ is bounded by [Etter, 1981]:

$$0 < \mu_\Delta < \frac{1}{10P_x} \quad (2.42)$$

where P_x is the power in the input signal. The ADF algorithm finds its main use when the adaptation rate for the LMS algorithm is required to be small in order to minimize the misadjustment. In the LJC experiment, due to the causality issue mentioned above, the experimental values of ϕ_G results in a *Lead* angle for frequencies above about 10 Hz. In

practice, Δ_G is started out with a delay of about one period. In this manner, the ADF algorithm can then adapt to a "lead" angle, thus correcting the problem of ϕ_p being less than ϕ_S.

2. Effect of Bias on the LMS Filter

In most laser targeting schemes, some sort of compensator is used to correct the bias error at the target. A second controller is then used to remove the "noise" in the beam. It has been noted by Widrow that an adaptable bias weight may be used to counteract the effect of "plant drift" in an adaptive inverse modeling situation [Widrow, 1985, pp. 283-284]. This single adjustable weight adapts to remove any bias in the output of the plant. It was noted during testing on the LJC, that by adding bias to the *reference signal*, the LMS filter adapted more quickly to the disturbance, and that the signal was zeroed on the target, without the use of any compensator. This effect may be analyzed as follows. Referring to Figure 4, let $v(n) = C + A\sin(2\pi fnT_s)$ where C is a bias error. Then the desired signal (the signal that must be cancelled) is:

$$d(n) = PC + PA\sin(2\pi fnT_s + \phi_p) \tag{2.43}$$

where *P* is the gain through the optical bench and mirrors and ϕ_p is the delay, as before. Now, considering only the bias effect, let the generated reference signal be:

$$x(n) = C' + B\sin(2\pi fnT_s) \tag{2.44}$$

where *C'* is an adjustable bias. The output of the FSM will be:

$$\begin{aligned} e(n) &= d(n) - s(n) \\ &= \left[PC + PA\sin(2\pi fnT_s + \phi_p)\right] - \left[SWC' + SWB\sin(2\pi fnT_s + \phi_W + \phi_S)\right] \end{aligned} \tag{2.45}$$

Rearranging to place the DC components together we find:

$$e(n) = \left[PC - SWC'\right] + \left[PA\sin(2\pi fnT_s + \phi_p) - SWB\sin(2\pi fnT_s + \phi_W + \phi_S)\right] \tag{2.46}$$

It is seen here, that the proper selection of *C'* will result in the cancellation of the DC component of the error. It is also noted, that if *C'* is zero, the LMS algorithm will be

unable to completely cancel the resulting error. As with the ADF, *C'* should be adaptable, since *C* may change and since *W* is adapting during the process to some quasi-steady state value.

a. The Adaptive Bias Filter

A similar process to the ADF is applied in the development of the Adaptive Bias Filter, or ABF. Using a one or two weight nLMS filter, the bias in the reference signal is adjusted to remove the DC component of the error signal. An estimate of *C'* is used as the reference signal to the ABF. The error signal in this case is the mean error, which stops the adaptation once the signal is centered on the target.

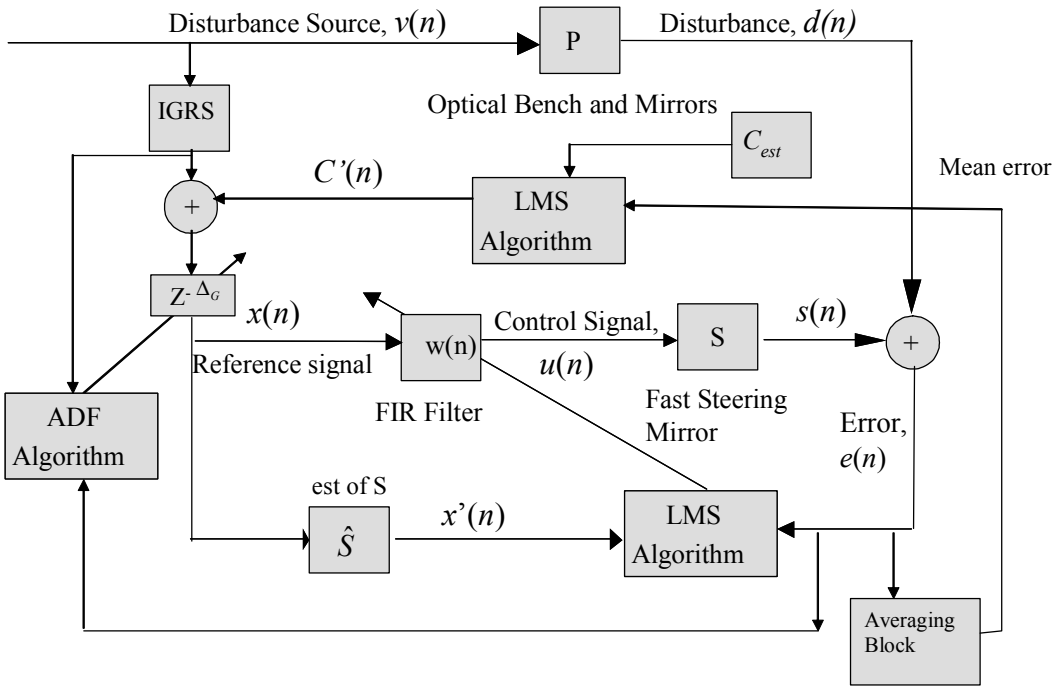


Figure 6 nFXLMS controller with ADF and ABF

C. THE GRADIENT ADAPTIVE LATTICE

The Gradient Adaptive Lattice (GAL) algorithm uses both forward and backward prediction to develop the estimate of the desired cancellation signal, which again drives the control mirror to cancel the effect of disturbance input. The construction of the GAL filter is different than the linear transversal filter in that it uses reflection coefficients vice tap gains due to the nature of the lattice-like structure as shown in the figure below.

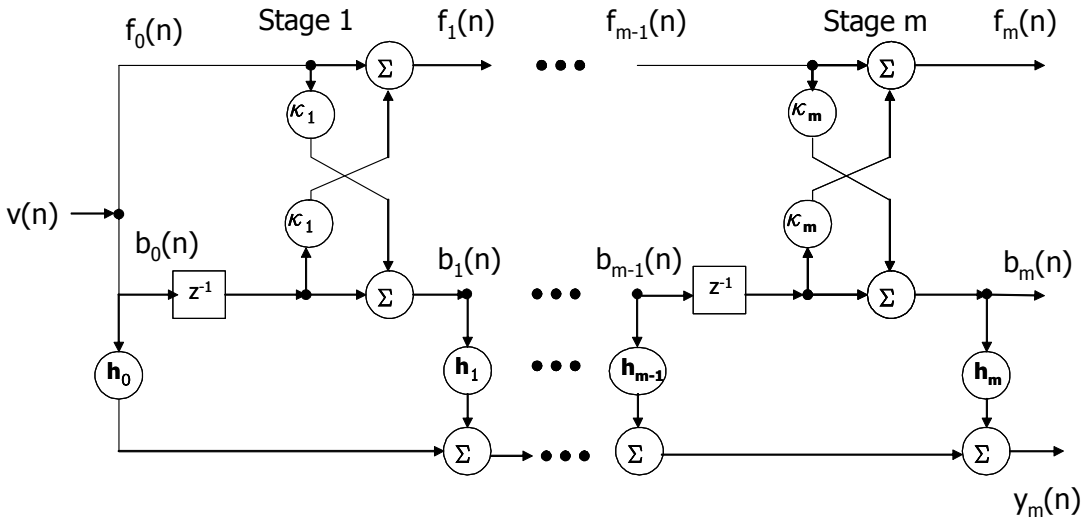


Figure 7 Gradient Adaptive Lattice [Haykin, 2002]

The forward prediction error for the m^{th} stage is given the symbol f_m(n). The backward prediction error, b_m is given likewise. κ is the reflection coefficient for the m^{th} stage and h_m provides the gains to construct the desired response, y_m. The h_m are calculated using the method of least squares, with the b_m as the regressor and the h_m as the parameters to be solved. The lattice filter results in a highly efficient structure in that it is modular. Each stage only requires the output of the previous stage, and an increase in the order of the filter only requires adding additional stages. Jiang, et al [Jiang, 1995] and Chen [Chen, 2001] discuss active noise cancellation techniques using algorithms that employ the lattice filter. The same concept was used in this study, but the algorithm from Haykin [Haykin, 2002, pp. 536-543] was used due to its simpler implementation. Although modular in its implementation, the GAL is computationally expensive, as the calculation of the reflection coefficients is very complex.

In brief, the lattice filter reflection coefficients are determined by minimizing the cost function:

$$J = \frac{1}{2} E\left[\left| f_m(n) \right|^2 + \left| b_m(n) \right|^2 \right] \tag{2.47}$$

Using the input-output relationships for the f_m and b_m one can determine the optimum κ by differentiating the cost function with respect to κ. This leads to the *Burg estimate* [Burg, 1968] for the values of the optimum reflection coefficient:

$$\hat{\kappa}_m(n) = -\frac{2\sum_{k=1}^{n} b_{m-1}(k-1) f_{m-1}^{*}(k)}{\sum_{k=1}^{n}\left(\left|f_{m-1}(k)\right|^2 + \left|b_{m-1}(k-1)\right|^2\right)} \tag{2.48}$$

where the * denotes conjugation. The determination of the h_m for the desired response estimator uses the normalized LMS algorithm, with the input being the backward prediction error, b_m, as the reference signal. It can be shown that the b_m are all orthogonal to one another [Haykin, 2002, pp172-173]. The advantage to the GAL is that since the b_m are all orthogonal, the input correlation matrix ($\mathbf{R}$) is a diagonal matrix, making the calculation of the h_m relatively simple, since by the Weiner-Hopf equations, $\overline{h} = \mathbf{R}^{-1}\overline{p}$. We can thus directly calculate the optimum tap weights for the desired response estimator. This allows for a rapid convergence of the algorithm to the optimum output.

Another feature of the GAL is its use of a time varying step size in the determination of the reflection coefficients. The step size is inversely proportional to the energy in the prediction error. A small value for the prediction error indicates that the predictor is providing an accurate model. This normally works well in that when the prediction errors are small, the algorithm uses a large step size and the filter can respond rapidly to changes [Haykin, 2002, p. 542]. However, if the input has a random component, as may be expected with a satellite laser relay system, the prediction errors become large and the algorithm responds much more slowly, which may be a detriment to the GAL in its use with the LJC in controlling "colored noise."

III. EXPERIMENTAL SETUP

A. LASER JITTER CONTROL TESTBED

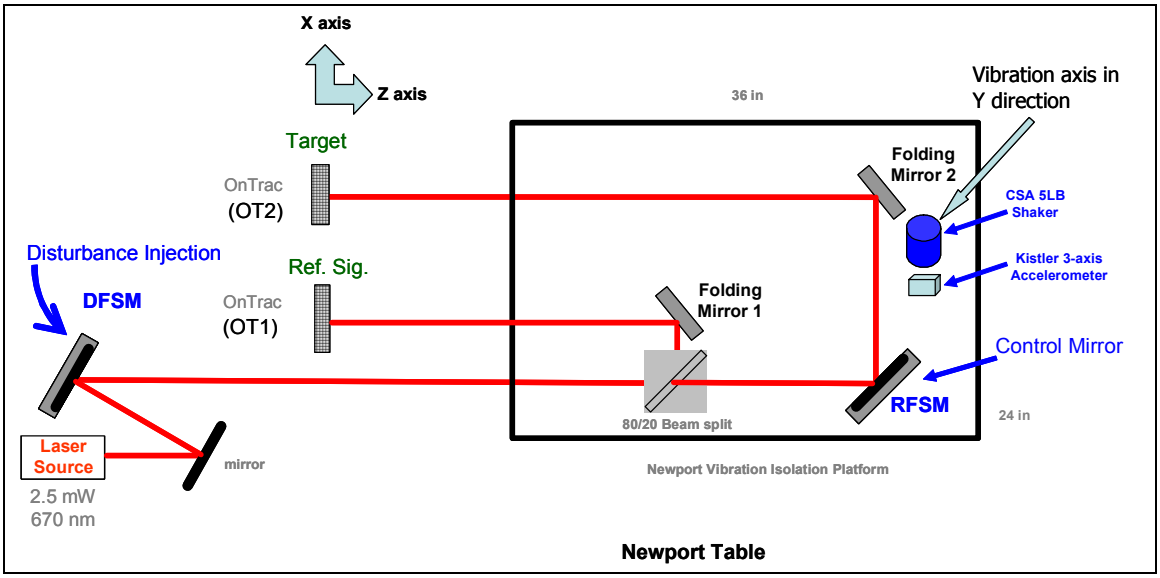


Figure 8 Laser Jitter Control Testbed

The Laser Jitter Control (LJC) Testbed is located in the Spacecraft Research and Design Center - Optical Relay Mirror Lab at the Naval Postgraduate School in Monterey CA. The components are mounted on a Newport Optical Bench which can be floated in order to isolate the components from external vibrations. The idea is to simulate a satellite or vehicle "bus" which houses an optical relay system. The laser beam originates from a source and passes through a Disturbance injection Fast Steering Mirror (DFSM). The DFSM corrupts the beam using random or periodic disturbances simulating the disturbances that might occur as a result of the transmitting station or the tip and tilt the beam may suffer as it passes through the atmosphere. A vibration isolation platform is used to mount the relay system, and to isolate the relay system from the optical bench. The relay system platform may be vibrationally disturbed by a 5 lb inertial actuator, simulating on-board running equipment such as control moment gyros, reaction wheels, cryo-coolers, etc. The incoming beam is split and reflected off the platform to a Position Sensing Detector (PSD) in order to obtain a reference signal that indicates the

on-board and injected disturbances. It is recognized that this is not a normal means to measure the disturbances as one would not be able to mount a PSD off the 'bus'. However, this reference PSD may be seen as simulating an on-board Inertial Measurement Unit (IMU), which is normally available in satellites with an optical payload. Using an accurate and sensitive IMU and a PSD, one could determine a similar reference signal. This setup allows a comparable measurement without the added cost of an IMU. A control FSM, designated the Receive FSM or RFSM, is used to correct the disturbed beam. The corrected beam is then reflected off the platform by a second folding mirror to the target PSD.

Figure 9 Laser Jitter Control Testbed

B. FAST STEERING MIRRORS

The Fast Steering Mirrors are the heart of the LJC. They are used to rapidly and accurately direct the laser beam through the system. The FSMs in the LJC use voice coils to position the mirrors in response to commanded inputs. The LJC uses two different FSMs, one by the Newport Corporation, and one by Baker Adaptive Optics.

1. Newport Fast Steering Mirror, Model FSM200

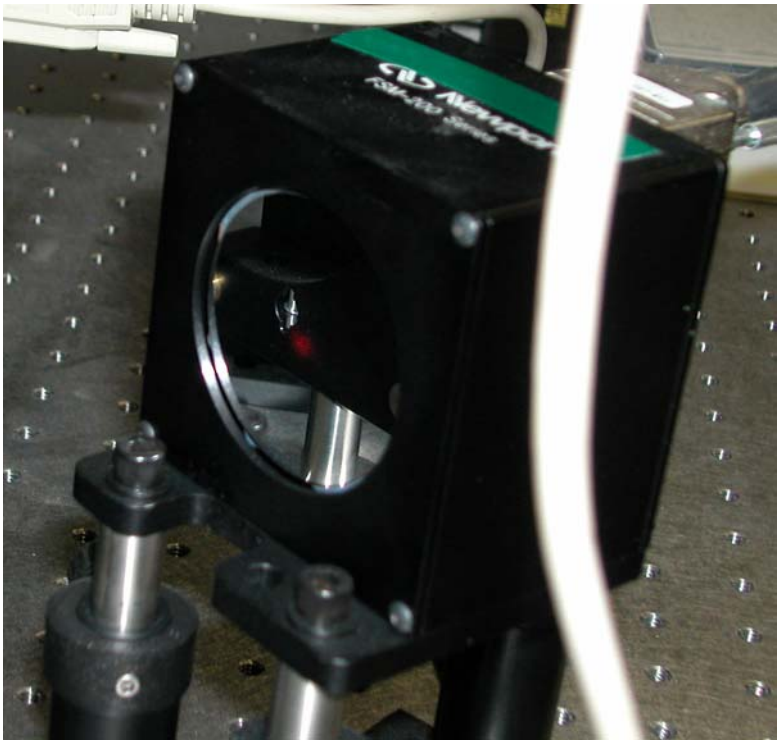


Figure 10 Newport Fast Steering Mirror

The Newport FSM is used as the control mirror (RFSM) for all experiments conducted during this research. The mirror comes with its own controller, the FSM-CD100, capable of both open loop and closed loop operation. The controller also provides an output of the mirror's angular position about each of the axis. In these experiments, the controller is configured in the open loop mode, with control inputs provided from the computer control system. The FSM200 is equipped with a two inch diameter mirror. The mirror substrate is mounted to its support by means of a flexure suspension, which does not require bearing surfaces, eliminating stiction and wear. The suspension for the FSM200 has a very small spring constant to minimize the current (and the subsequent heating) necessary to move the mirror. As a result, the mirror "hangs" when in the powered-off condition, and must be powered on before aligning the laser beam. It also must be powered on before shaking the platform to prevent damage to the suspension system. The actuator voice coils are mounted behind the mirror, and operate

in a push-pull manner, rotating the mirror about the axis that bisects each pair. The axes of the FSM200 are aligned with the horizontal and vertical directions as referenced to the mirror housing. The FSM has a control bandwidth of about 800-1000 Hz [Newport, 2002]. The manufacturer's specifications and operating limits are provided in Appendix A.

2. Baker Fast Steering Mirror, Model "Light Force One"

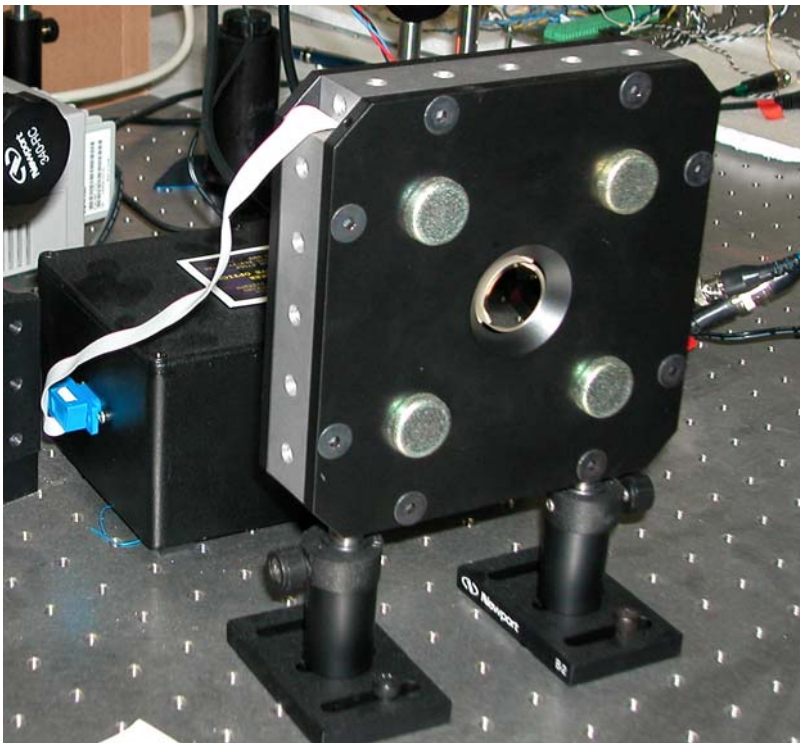


Figure 11 Baker Fast Steering Mirror

The second FSM used in the LJC is from Baker Adaptive Optics. The Light Force One is a one inch diameter mirror used for disturbance injection (DFSM) in these experiments. The Baker mirror uses a stiffer spring to mount the mirror substrate, which allows the mirror to remain in position when powered off. The Baker mirror comes with a small driver for positioning the mirror; however there is no closed loop option and mirror position is not available. The control bandwidth for the Baker mirror is about 3 kHz. Specifications are provided in Appendix A.

C. POSITION SENSING DETECTORS

Figure 12 Position Sensing Module

The laser beam optical position sensors, known as Position Sensing Modules or PSMs, were purchased from OnTrak Photonics Inc. and have a detection area of 10mm x 10mm. Each duolateral, dual axis PSM (model PSM2-10) requires an amplifier, the OT-301 to output the x and y position (in volts) of the centroid of the laser beam spot. The combination of amplifier and detector is called a Position Sensing Detector (PSD). Various beam splitters and folding mirrors are used to direct the beam as desired. The detectors have a minimum sensitivity of 50-100 μW, which drives the selection of the beam splitters and the determination of laser power. Their frequency response for the gain settings normally used is 15 kHz. The optimum beam diameter for accurate detection on the PSM is between 1 and 3 mm, and the maximum allowed intensity should be less than 300 W/cm^2. The output range of the PSM2-10/OT-301 detector is ±10 V and the OT-301 amplifier has a noise level of 1 mV. The minimum resolution of the PSM2-10 with the OT 301 amplifier is therefore 0.5 μm [Hunnicutt, 2003]. The specifications and drawings are provided in Appendix A.

D. NEWPORT VIBRATION ISOLATION PLATFORM

Figure 13 Newport Vibration Isolation Platform

The RFSM, beam splitters and folding mirrors are mounted on a Bench Top Pneumatic Newport Vibration Isolation Platform. This platform allows the control system actuators and optical path to be vibrated independent of the Optical Bench. The breadboard, which is self-leveling, rests on four pneumatic isolators.

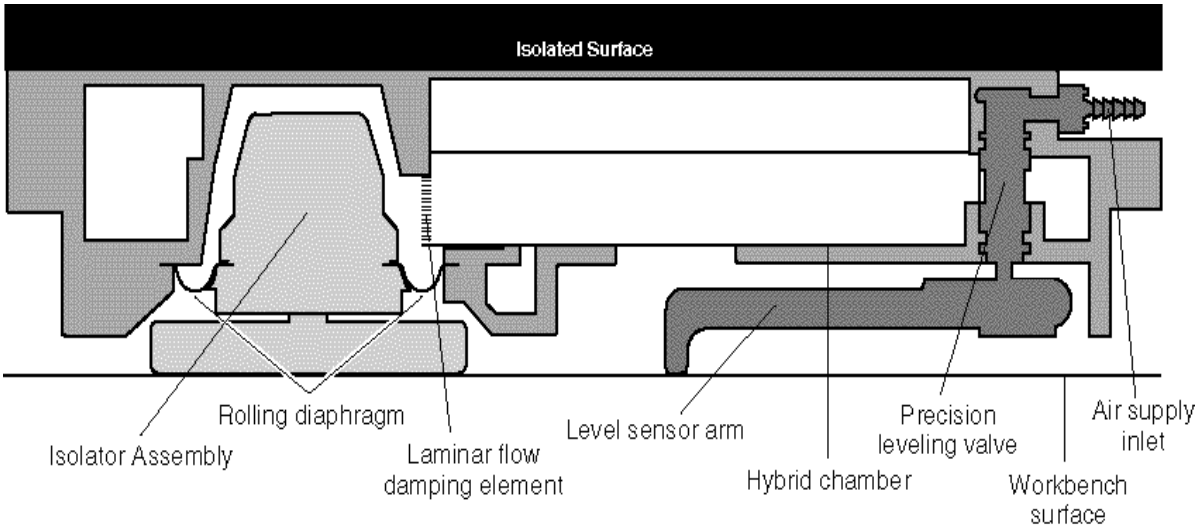


Figure 14 Platform Pneumatic Isolator Assembly and self-leveling system

The response of the platform to vibrations from the *workbench surface* is shown below:

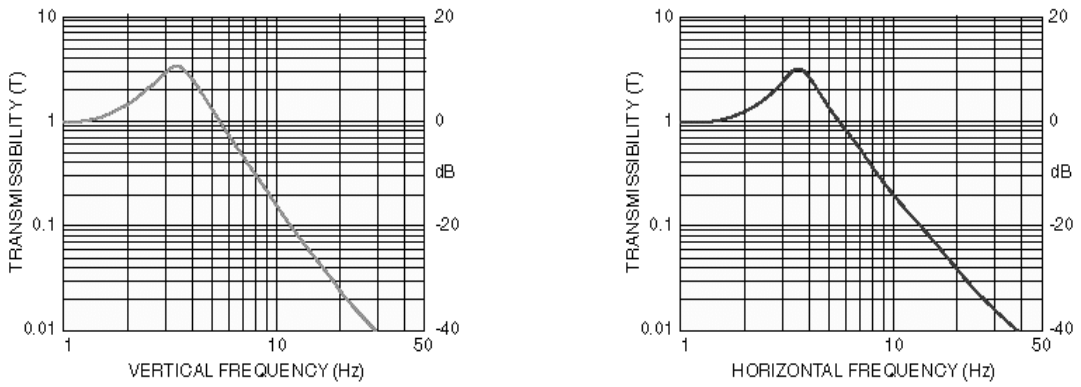


Figure 15 Platform Frequency Response

As noted above, the Isolation Platform was designed to isolate the breadboard from the vibrations of the workbench surface. In our experiment, we wished to vibrate the breadboard, and isolate the workbench surface from the vibrations. This concept was discussed with the Newport technicians, and they agreed that the system would work in reverse, as long as the vibrations were of such small amplitude as to not cause the self-leveling system to attempt to correct. The vibrations we intended to induce were on the order of 10 to 500 microns, which was well outside the performance specifications of the leveling system, and thus it would not correct the tilt of the platform induced by the mounted inertial actuator.

E. CSA ENGINEERING INERTIAL ACTUATOR

In order to vibrate the platform at desired frequencies, an inertial actuator is mounted on the vibration isolation platform. This actuator is a CSA model SA-5, capable of delivering a rated force of 5 lbf, in a bandwidth of 20 to 1000 Hz. The SA-5 has a resonance at about 40 Hz. The "shaker" may be mounted in any orientation. For our experiments, it was mounted next to the last folding mirror before the beam exits the platform, in the vertical direction, to provide maximum x and y axis motion. A separate bracket was designed to allow mounting in the horizontal direction as well. The manufacturer's specifications and drawings are provided in Appendix A.

F. KISTLER 3-AXIS ACCELEROMETER

A Kistler model 8690C10 accelerometer is employed to measure the disturbances generated by the shaker on the platform. The accelerometer is driven by a Kistler Piezotron Coupler, model 5124A.

G. COMPUTER CONTROL SYSTEM AND SOFTWARE

The computer control system is based on MATLAB, version 6.1 release 13 with SIMULINK, and the xPC Targetbox, all from the Mathworks. The main computer for control implementation and experiment supervision is a 2.4 GHz Dell Dimension 8250 with 1 Gbyte of RAM. The xPC Targetbox has the ability to accept 16 differential inputs and provide 4 analog outputs. The Targetbox is a Pentium III class computer running at 700 MHz. The Disturbance Computer, currently controlling the shaker, uses an AMD Athlon processor running at 1.4 GHz, with 256 Mbyte of RAM. dSPACE ver 3.3 with ControlDesk ver 2.1.1 is used to interface with the shaker. Future plans envision the use of two control mirrors. This will require the disturbance computer to control the DFSM as well as the shaker, in order to free up analog outputs from the xPC Targetbox.

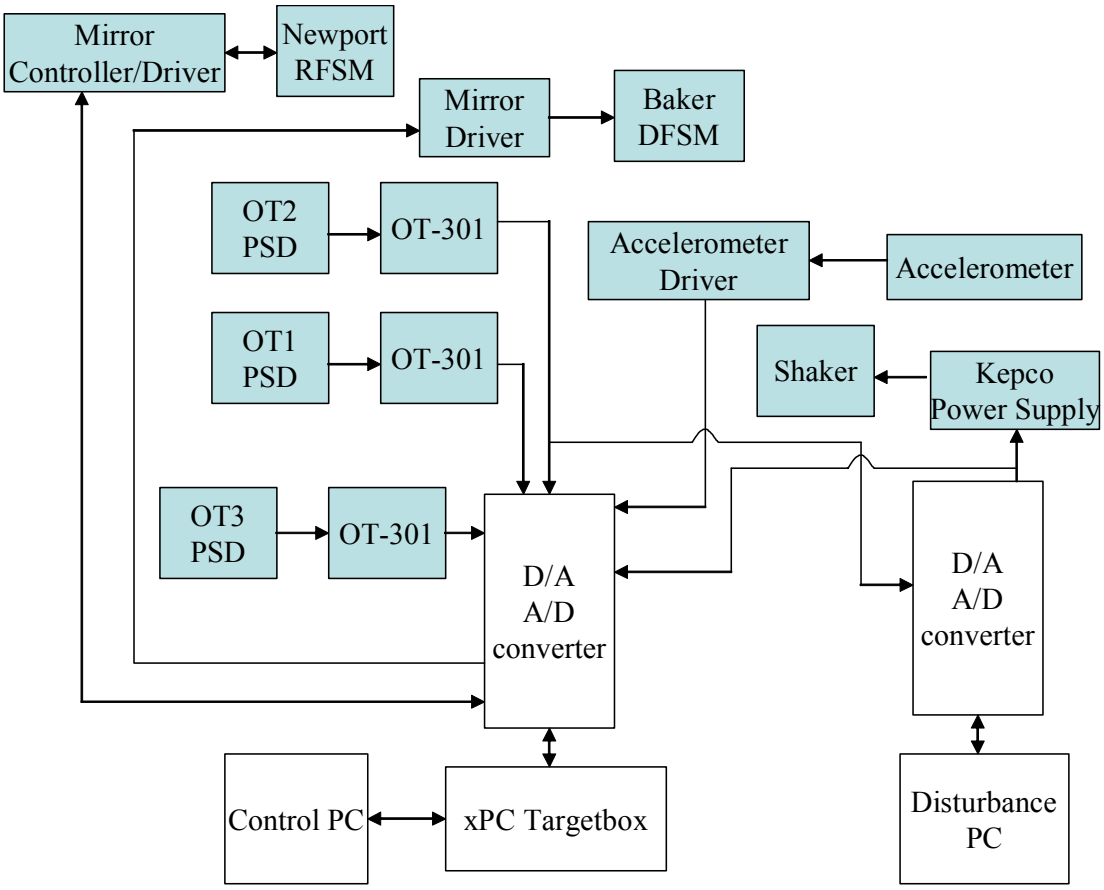


Figure 16 Signal Flow Diagram

The Control PC is used for several functions. Coding for control algorithms is performed in the MATLAB/SIMULINK environment. Some specialized functions, such as the lattice algorithm, require SIMULINK "S-functions". These are coded in C and compiled using the installed C++ compiler. xPC Target software compiles, generates code and downloads to the Targetbox the SIMULINK model for use in the control scheme. The Targetbox then runs the code automatically. The control PC provides supervisory functions during code execution, and allows user interface to the running code through the SIMULINK model. The disturbance computer uses the ControlDesk software to input the desired frequency, multiple frequencies, or random signals to the shaker through a Kepco Power supply. The output of the target PSD, OT2, is also routed to the disturbance PC through the dSPACE interface to provide a real-time plot of the beam position. The versions of software used for the LJC are listed in Appendix B.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. SYSTEM IDENTIFICATION EXPERIMENTS AND CALIBRATION

A. SAMPLE RATE

The frequencies of interest for the control of optical beam jitter vary from band-limited white noise at up to 1 kHz, to multiple narrow band frequencies in the range of 10 to 200 Hz. Due to the control bandwidth of the Newport mirror (800 Hz) and some trial runs, we determined that the mirror could handle up to one-fourth of its control bandwidth and maintain a reasonable control effort (i.e. not exceed the mirror's current limitations). We therefore chose to limit the random input to 200 Hz band-limited white noise. Since we were attempting to control a "noisy" signal, we decided to use the fastest sampling rate possible without overloading the CPU. Once the control algorithms and SIMULINK models were designed, we determined a sample rate of 2 kHz was possible. It is unlikely that any noise or frequency components lie above the Nyquist frequency of 1 kHz for our system. Additionally, any filtering of the input signal will necessarily incur a phase delay, which in the case of random noise at 200 Hz could adversely affect the ability to mitigate the jitter in the optical beam. For these reasons, an anti-aliasing filter was not used.

B. STEP, IMPULSE AND FREQUENCY RESPONSE OF THE FSM OPTICAL SENSOR SYSTEM

As a first attempt to model the system, Bode diagrams for the X and Y axis of the Newport FSM and Baker FSM were determined. The Baker plots used the voltage-in as the input, and the output was taken from a PSD located near the mirror, as the Baker mirror has no position sensor. The shape of the curve, especially the response at frequencies above 1 kHz, was discussed with the manufacturer, who reported that the response was typical of the mirror as designed. The speed of response of the Baker mirror is such that a very short (less than 4 microseconds) "spike" in the output of the D/A converter of the xPC Targetbox is picked up by the Baker mirror. The spike occurs at the "zero-crossing" of the signal, and is believed to be a problem with the circuit board in the Targetbox.

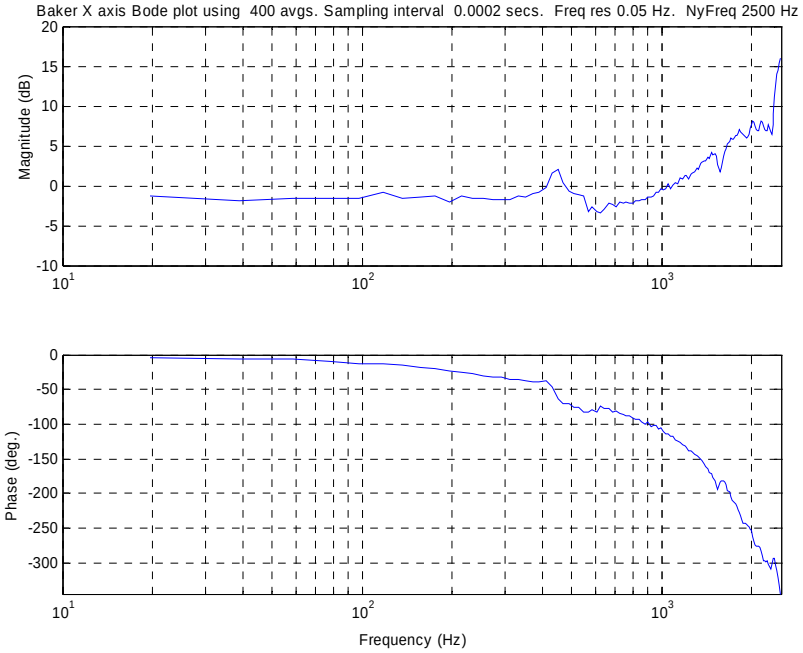


Figure 17 X axis Bode Plot of the Baker Mirror

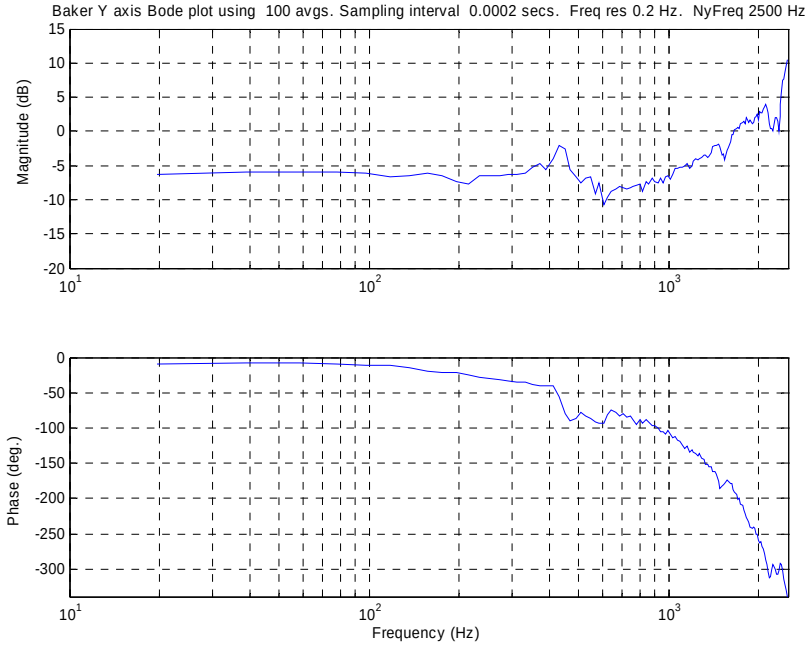


Figure 18 Y axis Bode Plot of the Baker Mirror

The spike currently prevents effective use of the Baker Mirror as a control mirror, and forces its use as the DFSM. The Newport mirror does not pick up this spike, and is not affected by it. This is probably due to the lower bandwidth of the Newport mirror.

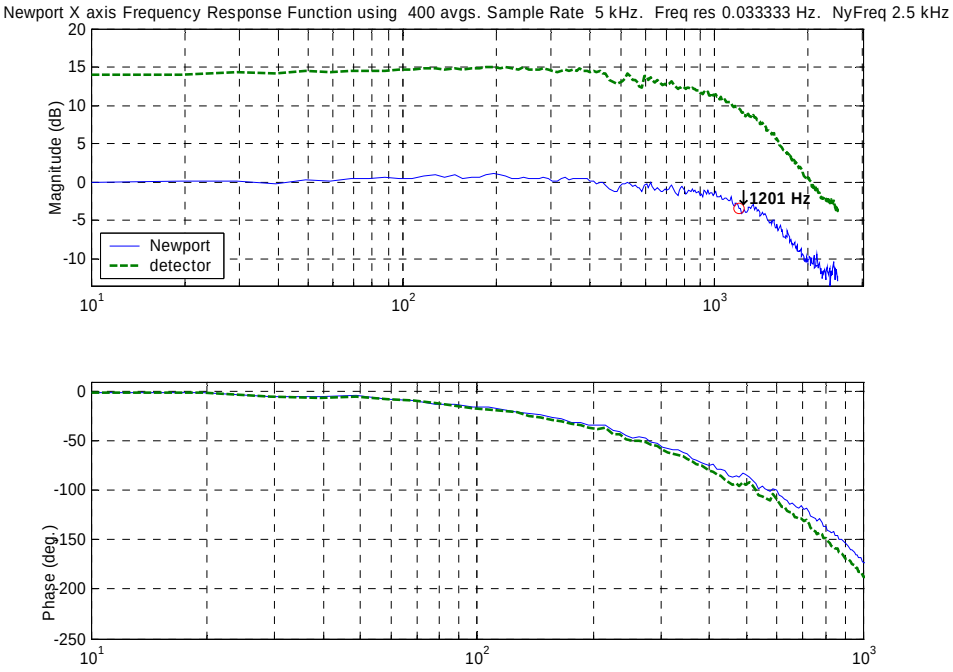


Figure 19 X axis Bode Plot of the Newport Mirror

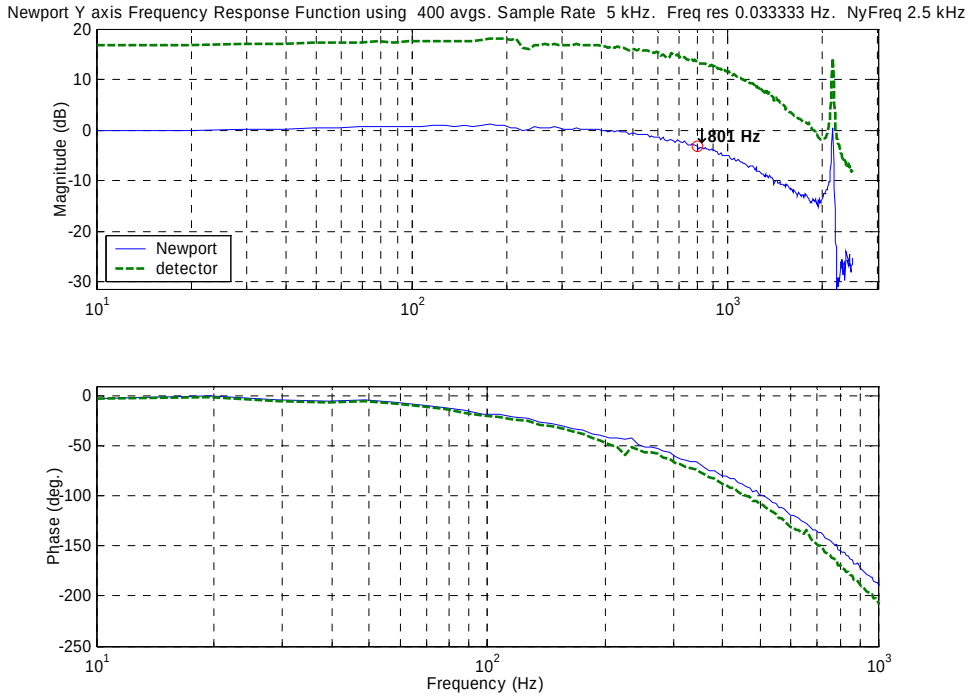


Figure 20 Y axis Bode Plot of the Newport Mirror

The Newport Bode plots have the 3 dB down points marked to show the bandwidth of the mirror. The detector output is plotted also, indicating the response of the detector. The detector curve shows the effect of the additional pole by comparing the slope of the magnitude line for the mirror with that of the detector at frequencies above 1 kHz. The responses for the Newport mirror are typical of that for a highly damped, 2nd order system.

Using the frequency, step, impulse and sinusoidal responses of the Newport mirror (RFSM) and target detector, a State-Space model was determined using classical analysis, similar to that given in Ogata [Ogata, 2002, pp. 224-233]. The parameters were adjusted to give the best fit for the sinusoidal response in the frequency range of 50 to 100 Hz, since we are primarily concerned with the ability of the adaptive systems to manage the vibrations of the platform. The State-Space model was constructed to allow for cross-coupling between axes. A comparison of the model to the actual system is given below for the X axis of the RFSM.

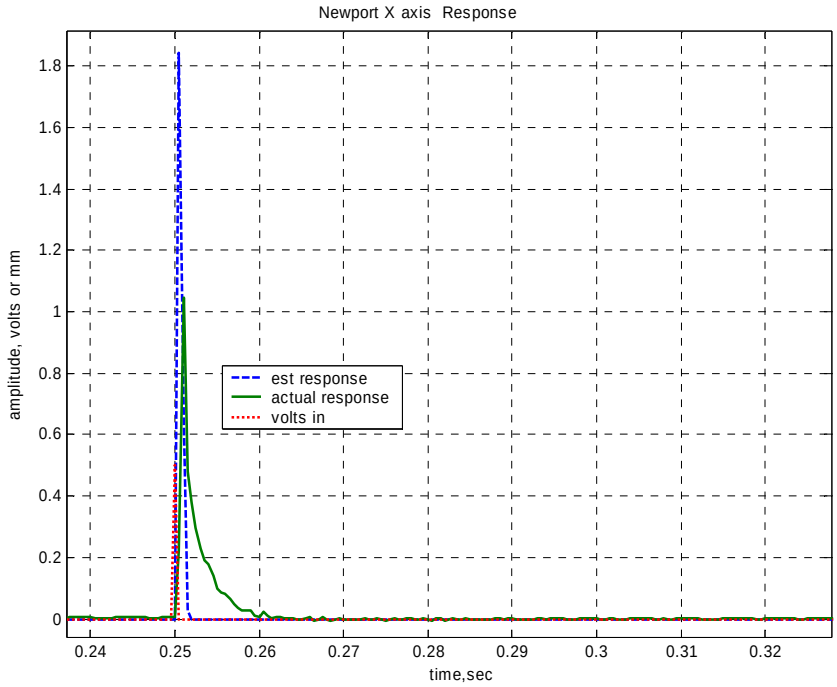


Figure 21 Impulse Response, Newport X axis

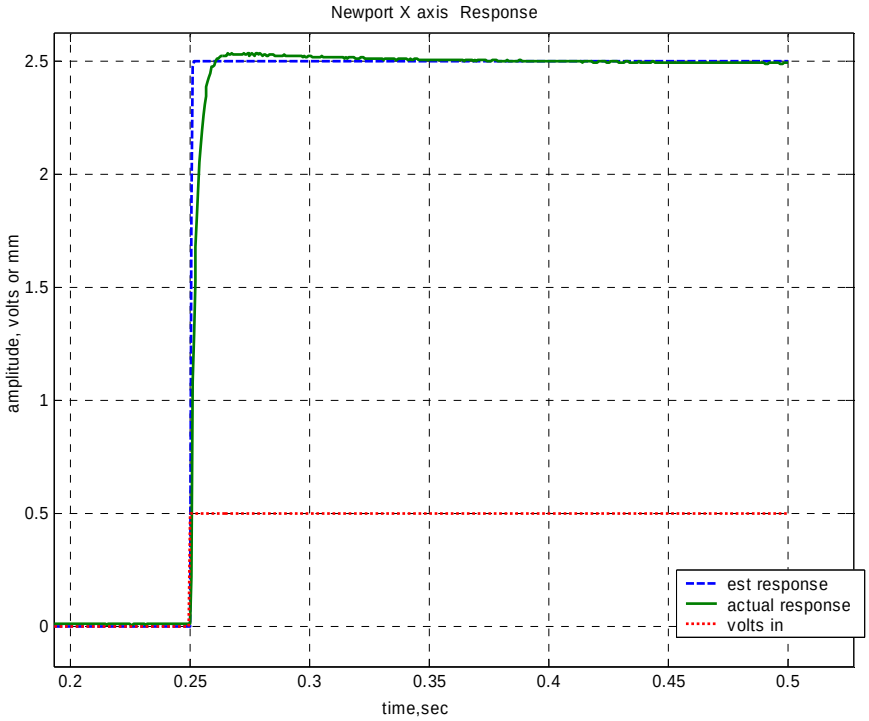


Figure 22 Step Response, Newport X axis

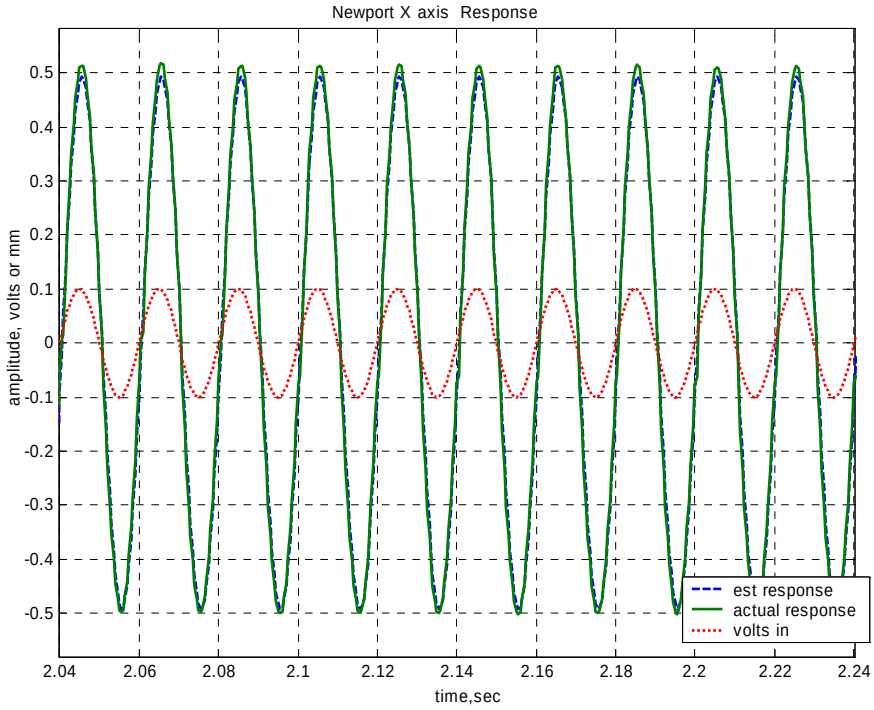


Figure 23 50 Hz Sinusoidal Response, Newport X axis

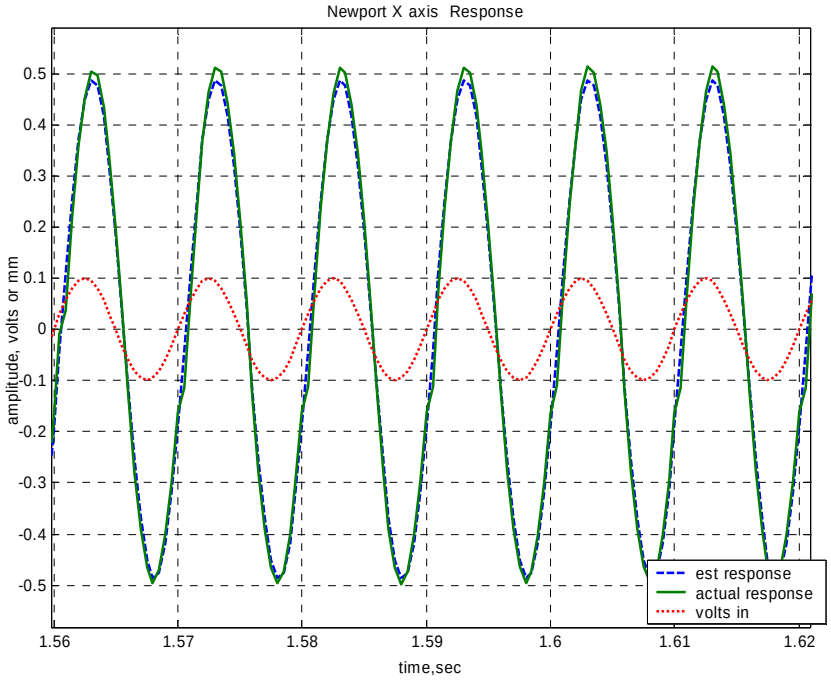


Figure 24 100 Hz Sinusoidal Response, Newport X axis

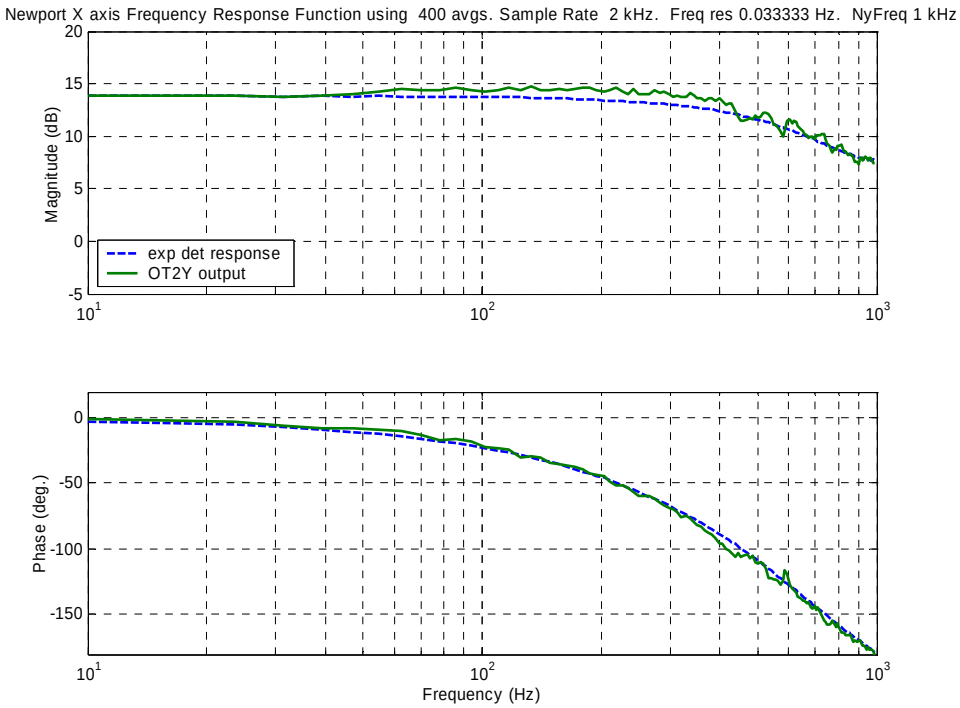


Figure 25 Frequency Response, Newport X axis

Although the impulse and step response of the model do not compare as favorably with the output from the mirror as may be desired, the parameters are determined with emphasis on the frequency response characteristics.

C. STATE-SPACE MODEL OF THE SYSTEM

Using equation (2.6), and expanding it to cover both axes, the State-Space model for the LJC Testbed is given below. The parameters used in the model response given in the figures above are provided in Table 1

$$\begin{bmatrix} \dot{V}_{py} \\ \dot{\theta}_x \\ \ddot{\theta}_x \\ \dot{V}_{px} \\ \dot{\theta}_y \\ \ddot{\theta}_y \end{bmatrix} = \begin{bmatrix} -1/T & 2gt_{\theta y}Gp_yDm/T & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -\omega_x^2 & -2\zeta_x\omega_x & 0 & A_x\omega_x^2 & A_x 2\zeta_x\omega_x \\ 0 & 0 & 0 & -1/T & 2gt_{\theta x}Gp_xDm/T & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & A_y\omega_y^2 & A_y 2\zeta_y\omega_y & 0 & -\omega_y^2 & -2\zeta_y\omega_y \end{bmatrix} \begin{bmatrix} V_{py} \\ \theta_x \\ \dot{\theta}_x \\ V_{px} \\ \theta_y \\ \dot{\theta}_y \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ G_m\omega_x^2 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & G_m\omega_y^2 \end{bmatrix} \begin{bmatrix} V_{mx} \\ V_{my} \end{bmatrix} \quad (4.1)$$

$$\begin{bmatrix} V_{py} \\ V_{px} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} V_{py} \\ \theta_x \\ \dot{\theta}_x \\ V_{px} \\ \theta_y \\ \dot{\theta}_y \end{bmatrix} \quad (4.2)$$

Name	**Variable**	**Value**
PSD Voltage, Y axis	V_{py}	-10 to +10 volt
PSD Voltage, X axis	V_{px}	-10 to +10 volt
Rotation about X axis of RFSM	θ_x	-26.2x10^{-3} to +26.2x10^{-3} rad
Rotation about Y axis of RFSM	θ_y	-26.2x10^{-3} to +26.2x10^{-3} rad
PSD response Time	T	67 x10^{-6} sec
Geometry factor, Y axis	$gt_{\theta y}$	0.405 (typical)

Name	Variable	Value
Geometry factor, X axis	$gt_{\theta x}$	0.535 (typical)
PSD Calibration, Y axis	Gp_y	$2x10^3$ V/m
PSD Calibration, X axis	Gp_x	$2x10^3$ V/m
Distance from RFSM to Target	Dm	1.245 m
RFSM Damping Ratio, X axis	ζ_x	0.90
RFSM Damping Ratio, Y axis	ζ_y	0.90
RFSM Bandwidth, X axis	ω_x	5655 rad/sec (900 Hz)
RFSM Bandwidth, Y axis	ω_y	5184 rad/sec (825 Hz)
Mirror Calibration Factor	Gm	$52.4x10^{-3}$ rad/volt
Voltage input to RFSM, X axis	V_{mx}	-10 to +10 volt
Voltage input to RFSM, Y axis	V_{my}	-10 to +10 volt
Cross-coupling Factor, X axis	A_x	$-2\ x10^{-2}$
Cross-coupling Factor, Y axis	A_y	$-2\ x10^{-2}$

Table 1 List of Variables for the State Space Model

In equation (4.1), the geometry factors, $gt_{\theta y}$ and $gt_{\theta x}$ are used to correct for the 45 degree angle the RFSM is mounted at with respect to the incoming beam, as well as the alignment of the folding mirrors and beam splitters. The geometry factors vary depending on the height of the vibration isolation platform, which varies slightly with pressure, but is measurable on the PSDs. The cross-coupling factor was chosen by comparison of the output of the opposite axis with input on one axis. The RFSM

bandwidth and critical damping ratio were chosen by comparison of the model output with the actual frequency response (Bode diagram) and the sinusoidal response in the frequency range of interest (50 to 100 Hz). All other numbers are from manufacturers data. The factor "-1" in equation (4.2) takes into account the wiring difference between the feedback detector (OT3) and the target (OT2) and reference signal (OT1) detectors. OT3 faces OT2 and OT1 and therefore its X axis is reversed in direction from OT2 and OT1.

D. FREQUENCY RESPONSE OF THE PLATFORM

As part of the investigation into controlling the vibrating platform, we need the frequency response of the RFSM/platform system. A linear chirp signal was used to vibrate the inertial actuator, from 10 to 500 Hz, with a period of 1 sec and amplitude of 1 volt peak to peak. The Power Spectral Densities calculated in this research are all determined by the *pwelch* function in MATLAB which uses a Hamming Window to shape the data. The Y axis response on the PSD (rotation of the X axis of the mirror) is shown below:

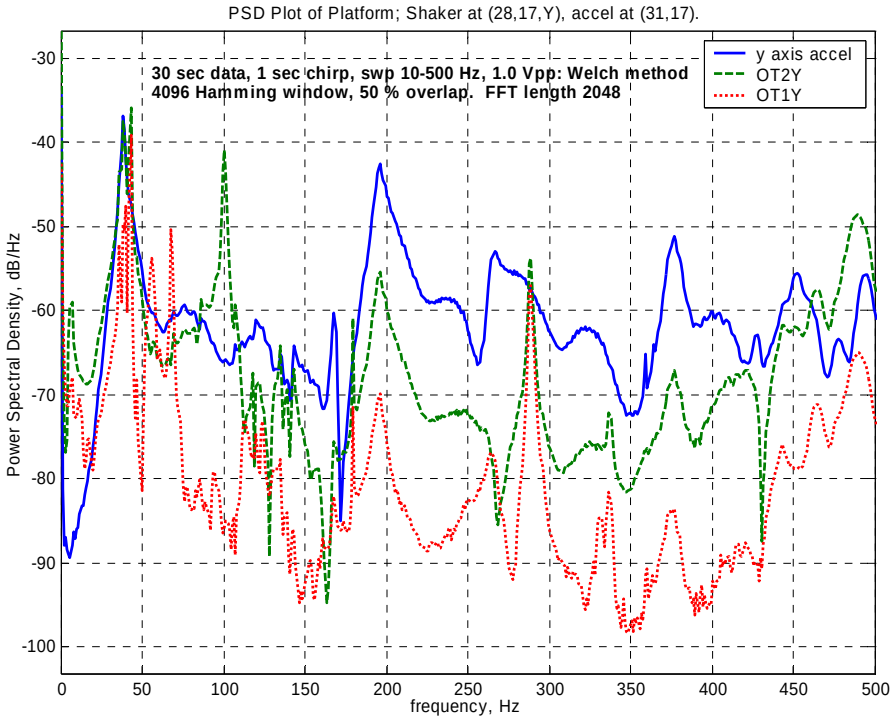


Figure 26 Power Spectral Density of the Platform

Figure 26 shows the response of an accelerometer placed next to the inertial actuator and the sensor response at both the target (OT2Y) and reference (OT1Y) PSDs. Power Spectral Density Plots were obtained using accelerometers at the top and bottom of both the Newport mirror and the last folding mirror on the platform, labeled FM2 and compared to the accelerometer located next to the inertial actuator.

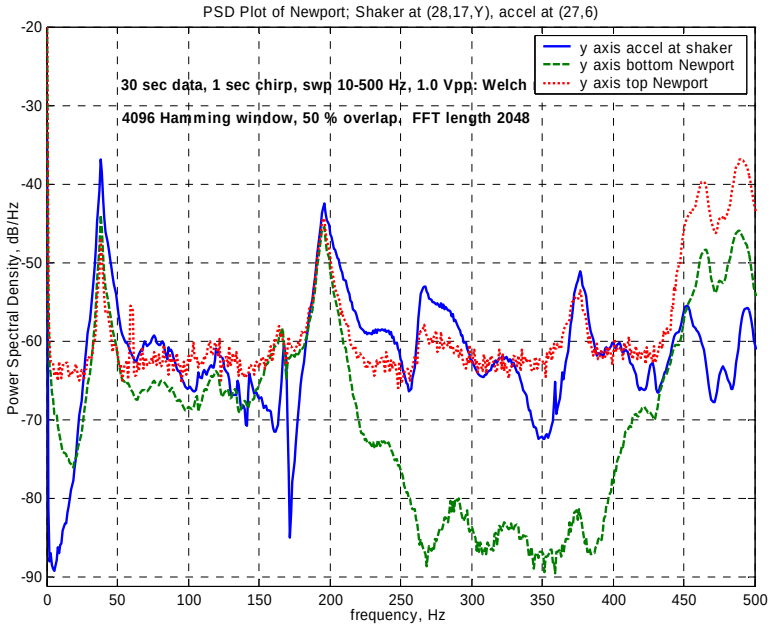


Figure 27 Power Spectral Density of the Newport Mirror

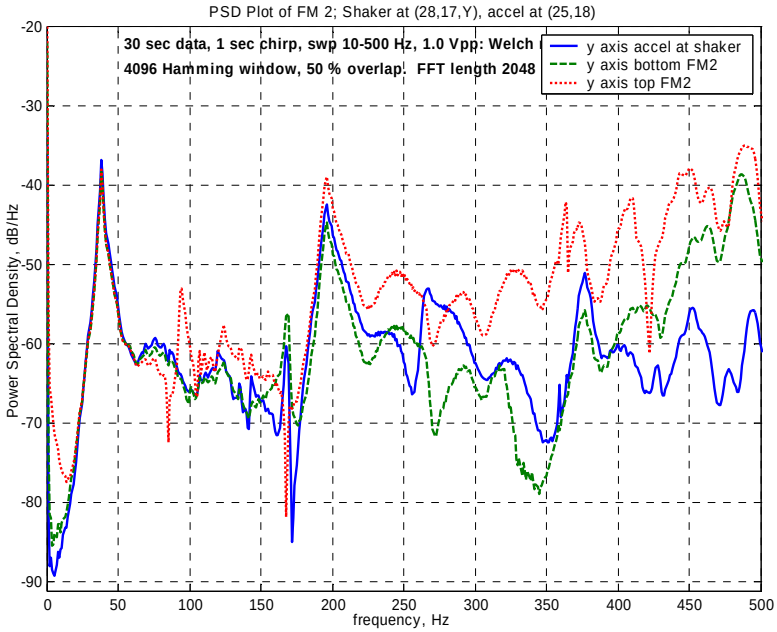


Figure 28 Power Spectral Density of Folding Mirror 2

For the frequencies of concern, approximately 50 to 100 Hz, we note that there are three major areas of interest. The peak at about 40 Hz in all three figures above is related to the shaker resonance. The peaks at 56 and 67 Hz (Figure 26) appear to come from Folding Mirror 1 (FM1) and only show up on the OT1 (reference signal) response. The peak near 100 Hz comes from FM2 (Figure 28). A look at the Frequency Response Function (FRF) using the control voltage to the shaker power supply as the input, and the sensor data (Y axis) as the output will serve to illustrate these points:

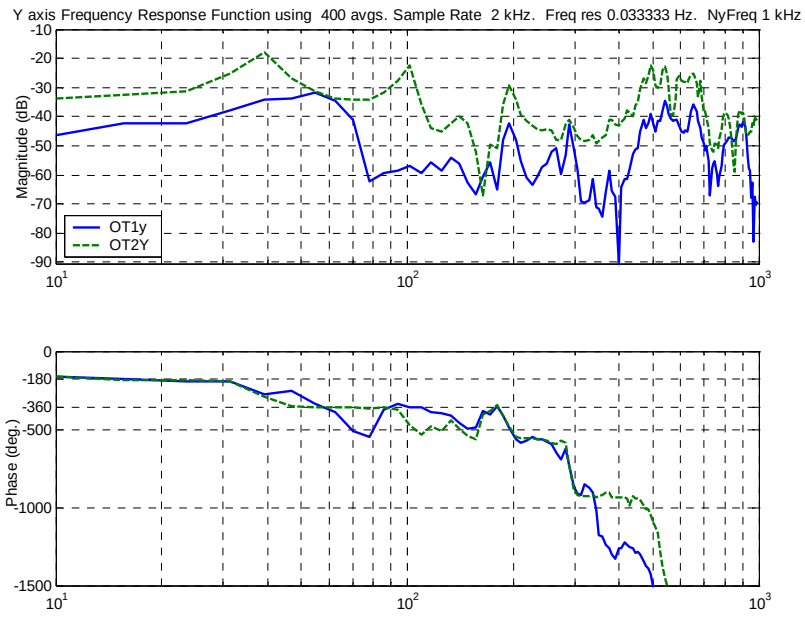


Figure 29 Frequency Response Function, shaker input to OT1Y, OT2Y

The FRF indicates that both target and reference PSDs develop a phase shift starting at 30 Hz. OT1Y, (reference signal Y axis) however, is about 90 degrees out of phase with OT2Y (target signal Y axis) between 40 and 50 Hz. OT1Y is 180 degrees out of phase with OT2Y between 70 and 80 Hz. As adaptive filters require a reference signal, these phase shifts will become important as we attempt to control the vibrations. It is noted, that even if one were to use the accelerometer located at the shaker as a reference signal, there is still a phase shift between the accelerometer and the target (30 to 50 Hz). The table below lists the significant frequencies excited between 0 and 200 Hz, and the probable source for each.

Frequency	Source
38 Hz	Inertial Actuator resonance, Folding Mirror 1 and 2,
40 Hz	Platform resonance
43 Hz	Folding Mirror 1 and 2
55 Hz	Folding Mirror 1
67.5 Hz	Folding Mirror 1
100 Hz	Folding Mirror 2
196 Hz	Platform

Table 2 Frequency Sources

1. Damping of the Inertial Actuator/Platform System

The damping of the Inertial Actuator/Platform System was determined using the half-power points of the main resonance at 40 Hz. Figure 30 below is the Frequency Response of the Platform to a swept sinusoid forcing function, 35 to 47.5 Hz, to the inertial actuator. An HP 33120 Wavetek Signal Generator was as the source for the inertial actuator. An HP 35665A Digital Signal Analyzer was used to analyze the signal. The output of the system is the laser position as recorded on the target PSD, OT2. The damping calculation is shown in Figure 31 and is approximately 0.20% at 40 Hz, which matches the Digital Signal Analyzer determination of the damping at the resonant frequency.

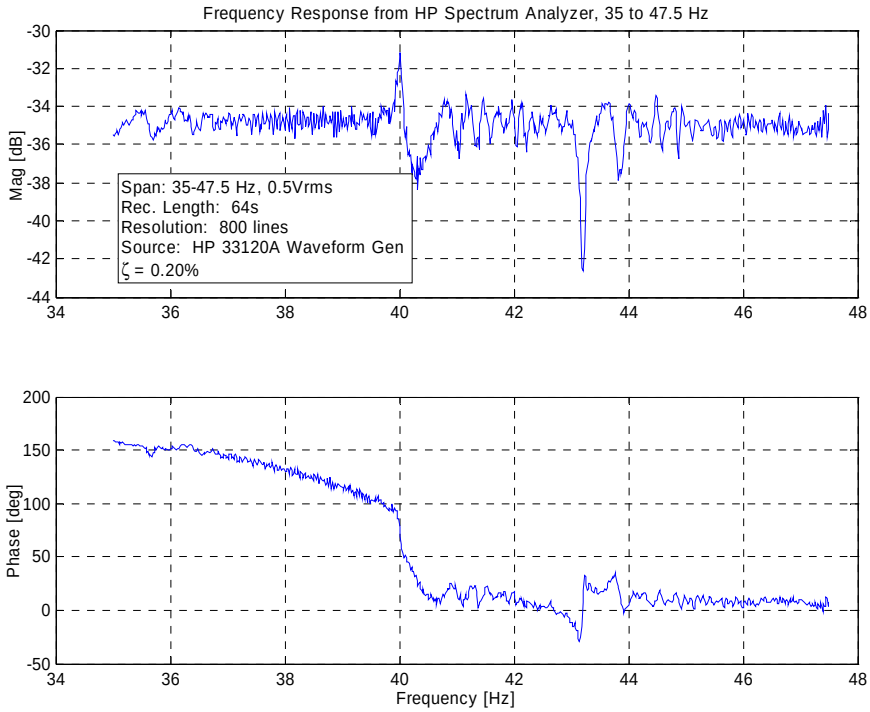


Figure 30 Damping of the Inertial Actuator/Platform System

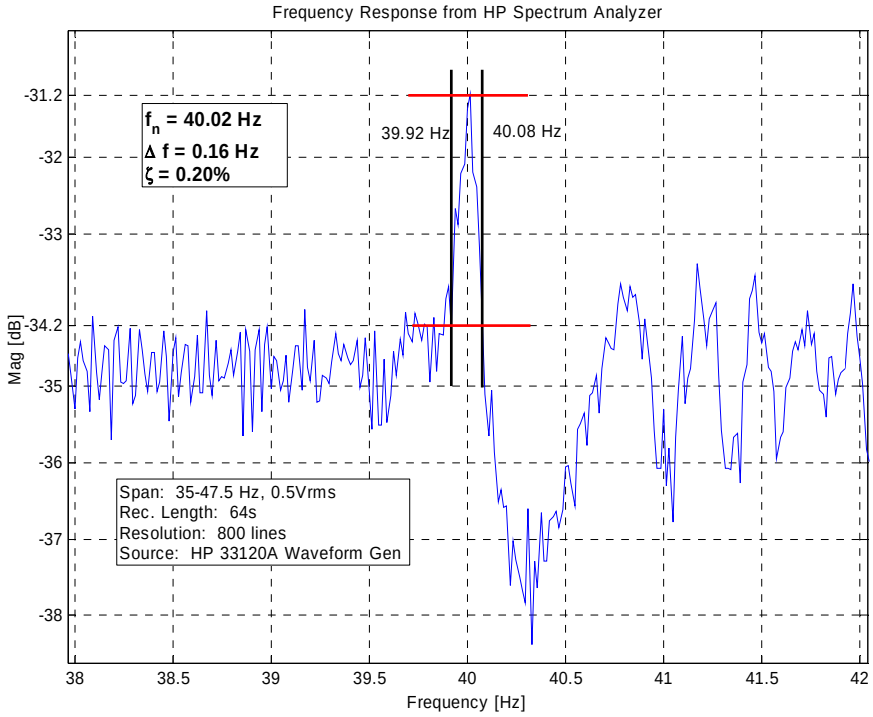


Figure 31 Damping Calculation

E. CALIBRATION

In order to use the LJC testbed, a calibration of the sensors and mirrors is necessary. Testing of the mirror's motion about both axis and the sensor output is required to verify operability of the system. Each day, before testing, a calibration program is run to check the system. If at any time during testing, unusual results are obtained, the calibration program is run again to check system response. The program may be run from any directory, simply by typing the command "calibrate". Once the calibration run is complete, the user may opt to keep the calculated gains for the system, or delete them if desired.

To calibrate the system, a zero input data run is made. This shows how close the alignment is to having the beams "zeroed" on the detectors. Since the incident angle of the beam on the RFSM and detectors is critical to the ability of the system to control the beam, this zero input data run allows the user to align the folding mirrors to place the beam as close to zero as possible, at least within 0.1 mm, with the mirrors commanded at zero angle. The height of the X axis of the detectors, and the center of the RFSM, has been precisely placed at the height of the laser beam. By aligning the folding mirrors in this manner, as much of the nonlinearities (due to the angle of the beam) as possible is taken out of the system. Once that is complete, each mirror is commanded to rotate about its X and Y axis from the negative to the positive limits. With the laser beam on, this sweeps the beam across the PSDs in the horizontal (X axis) and vertical (Y axis) direction. Graphs are constructed using the detector output, allowing the user to check the response of both the detector and mirror. Step inputs are given to each axis of the mirrors, to check the step response of the system. Next, the mirrors center the beam on each of the PSDs, and calculate the input necessary to "zero" the beam. Using the data generated during the run of the detector response, gain of the mirrors, and the zero position, a 1 mm box is drawn centered on each detector. The user can easily verify proper operation by observing the boxes drawn by the mirrors. The figures below show the result for OT2, and compare the drawn boxes for a properly aligned system, vs. a poorly aligned system. The figure on the left is the three PSD outputs for a zero control

signal to the RFSM and DFSM prior to "drawing" the boxes. The difference in the OT2 box and the OT3 box is due to the difference in distance between the target and feedback detector. The calibration program and typical output are provided in Appendix C. Figure 33 also shows the "noise" in the beam from the thermal variations in the room and from the laser source itself

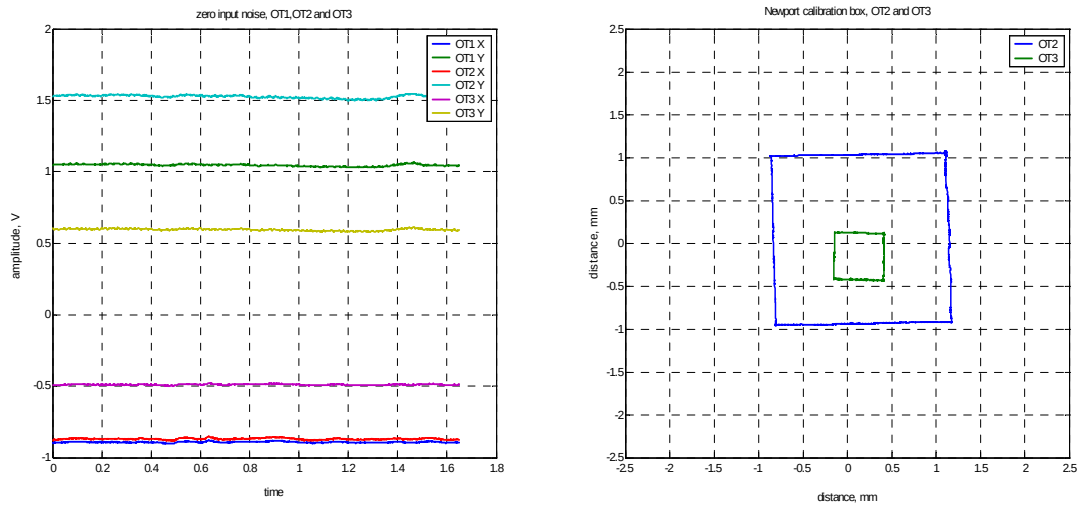


Figure 32 Poorly aligned Calibration Boxes

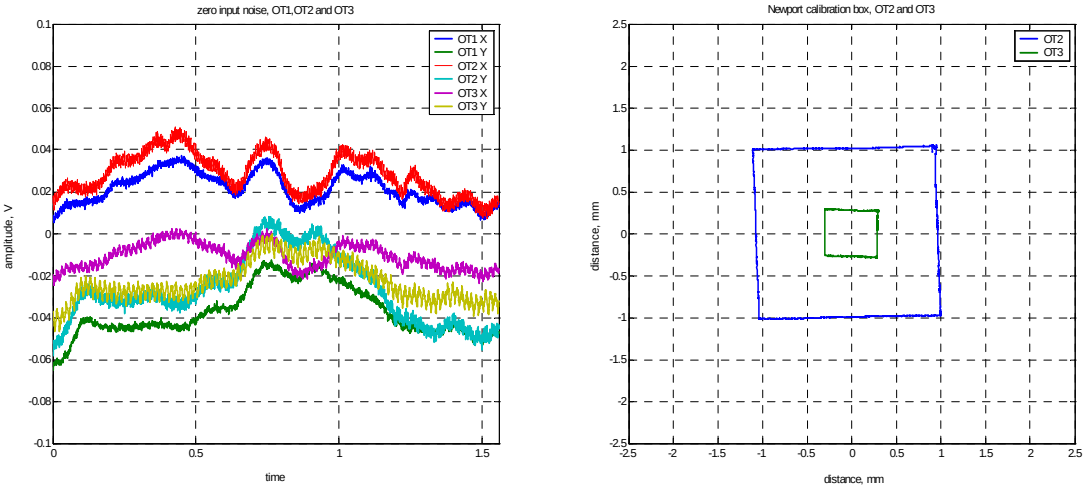


Figure 33 Properly aligned Calibration Boxes

THIS PAGE INTENTIONALLY LEFT BLANK

V. NUMERICAL SIMULATION OF THE CONTROL PROBLEM

Adaptive control worked well for the situation in which the platform containing the control system was stationary and the disturbance was introduced only by the DFSM. Once we began to induce vibrations on the platform however, there were several problems that appeared. The flexibility of the platform and associated mirrors affected the reference signal, causing degradation in the adaptive system's ability to remove the jitter in the beam. For example, the Gradient Adaptive Lattice (GAL) controller did not work well as frequency approached 50 Hz. For another, bias at the target was removed by the LMS controller, vice just passing this "DC signal" through. This bias, depending on the value, would severely impact the ability of the controller to remove even a single narrow band frequency. Although the mirrors and control system had been modeled in MATLAB, we did not see these effects in the simulation. In order to study these effects, we decided to devise a simple system, one that could be analytically solved and the simulations verified. The system should be straight forward, yet analogous to the control problem we faced.

A. THREE MASS SPRING AND DAMPER SYSTEM

A simple three mass spring and damper system was chosen, as shown below:

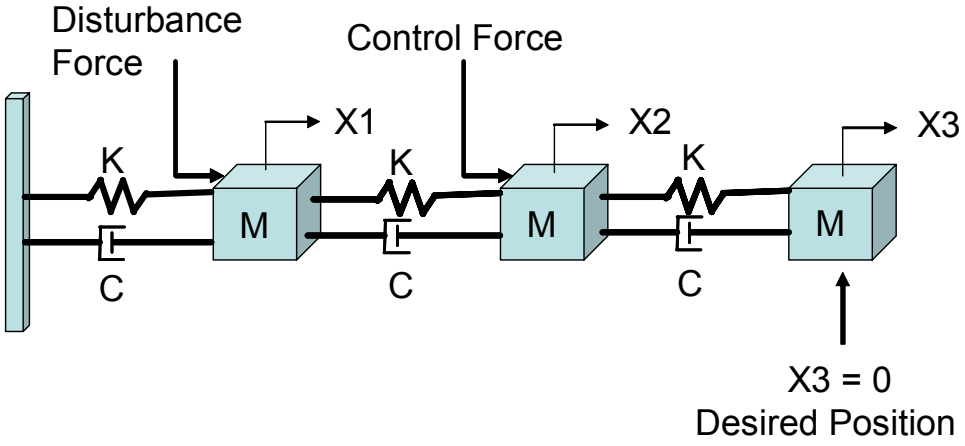


Figure 34 Three Mass System

As can be seen from the diagram, the disturbance force goes "though" the control point, similar to the effect on the testbed, in that the "disturbed" light beam goes "through" the RFSM. For this simple system, the value of *M*, the mass of the one-dimensional blocks, are all the same. Likewise, *K*, the spring constant and *C*, the damping coefficient, are the same, simplifying the analytical calculations. The positions of each mass are *X*1, *X*2 and *X*3 respectively. The control problem is to maintain mass 3 at the zero position, regardless of the force exerted at mass 1, by using a control force at mass 2. Since we are dealing with discrete systems in the experiment on the testbed, we decided to choose the value of *M* and *K* so that the first fundamental frequency would be similar to that we see on the platform. In a discrete system, the control algorithm must be calculated within the sample time of the system. We therefore use the same sample rate for the three mass system as for the experiment, namely 2 kHz.

B. ANALYTICAL SOLUTION

Since the values for the spring constants are the same, and the values for the damping coefficients are the same, the system exhibits proportional damping, allowing us to use modal decomposition to solve the coupled differential equations of motion. The modal solution can be shown to be:

$$q_i(t) = A_i \sin(\omega t + \phi_i) + e^{-\zeta_i \omega_i t}\left(C_{1i}\cos(\omega_{di} t) + C_{2i}\sin(\omega_{di} t)\right) \qquad i = 1,2,3 \tag{5.1}$$

where:

t is time
q_i is the i^{th} modal coordinate
ω_i is the i^{th} natural frequency
ζ_i is the i^{th} modal damping coefficient
ω is the forcing frequency, and
$\omega_{di} = \omega_i\sqrt{1-\zeta_i^2}$, the i^{th} damped natural frequency

Additionally,

$$\tan(\phi_i) = \frac{2\zeta_i \left(\frac{\omega}{\omega_i}\right)}{\left(\frac{\omega}{\omega_i}\right)^2 - 1} \tag{5.2}$$

$$A_i = \frac{\left(\frac{f_i}{\omega_i^2}\right)}{\sqrt{\left[1 - \left(\frac{\omega}{\omega_i}\right)^2\right]^2 + \left[2\zeta_i \left(\frac{\omega}{\omega_i}\right)^2\right]^2}} \tag{5.3}$$

where f_i is the i^{th} modal forcing function amplitude. Setting the initial conditions to zero, the constants C_1 and C_2 may be found:

$$\begin{aligned} C_{1i} &= -A_i \sin(\phi_i) \\ C_{2i} &= \frac{-A_i\left(\omega\cos(\phi_i) + \zeta_i\omega_i \sin(\phi_i)\right)}{\omega_{di}} \end{aligned} \tag{5.4}$$

Finally, mapping the modal coordinates to the coordinates of the system with the matrix of ortho-normal modes used to decouple the differential equations, we obtain:

$$\bar{x}(t) = \begin{bmatrix} X1 \\ X2 \\ X3 \end{bmatrix} = \left[\tilde{P}\right]\bar{q}(t) \tag{5.5}$$

where:

$$\left[\tilde{P}\right] = \left[\tilde{\phi}_1 \quad \tilde{\phi}_2 \quad \tilde{\phi}_2\right]$$

$\tilde{\phi}_i$ are the vectors of the ortho-normal modes

This system of equations can be used to solve for the motion of each of the masses, given an input force at mass 1 or mass 2 or a combination of both. The equations were programmed into MATLAB, to verify the accuracy of the MATLAB simulation of the three mass model.

C. NUMERICAL SIMULATION OF THE THREE MASS MODEL

For use in numerical simulations in MATLAB, the three mass model is posed in a State Space system of equations. Using standard dynamics equations the model is:

$$\begin{bmatrix} \dot{X}1 \\ \dot{X}2 \\ \dot{X}3 \\ \ddot{X}1 \\ \ddot{X}2 \\ \ddot{X}3 \end{bmatrix} = \frac{1}{M}\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -2K & K & 0 & -2C & C & 0 \\ K & -2K & K & C & -2C & C \\ 0 & K & -K & 0 & C & -C \end{bmatrix}\begin{bmatrix} X1 \\ X2 \\ X3 \\ \dot{X}1 \\ \dot{X}2 \\ \dot{X}3 \end{bmatrix} + \frac{1}{M}\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} Z \\ U \end{bmatrix} \qquad (5.6)$$

where Z is the disturbance input at mass 1 and U is the control input at mass 2. In order to obtain natural frequencies near the platform frequencies, we set $M = 20\ \frac{\text{lb}_\text{f}\text{sec}^2}{\text{in}}$ and $K = 2 \text{ x } 10^7\ \frac{\text{lb}_\text{f}}{\text{in}}$. These values give natural frequencies of 70.8, 198.5 and 286.8 Hz. C may be set to obtain the damping desired. The State-Space system of equation (5.6) is discretized using a sample rate of 2 kHz. A comparison of this model's output at mass 3 to the analytical solution for the motion of mass 3 for a damping of 5% of critical for the first mode ($C = 4420\frac{\text{lb}_\text{f}\text{sec}}{\text{in}}$) is shown in Figure 35, for a sinusoidal forcing function of 50 Hz. In the 3-mass model, the forcing function is implemented by the sine wave source block provided in the Digital Signal Processing Toolbox of MATLAB (DSP Sine Wave). The DSP Sine Wave is configured for multiple frequencies in the discrete mode, at a sample rate of 2 kHz.

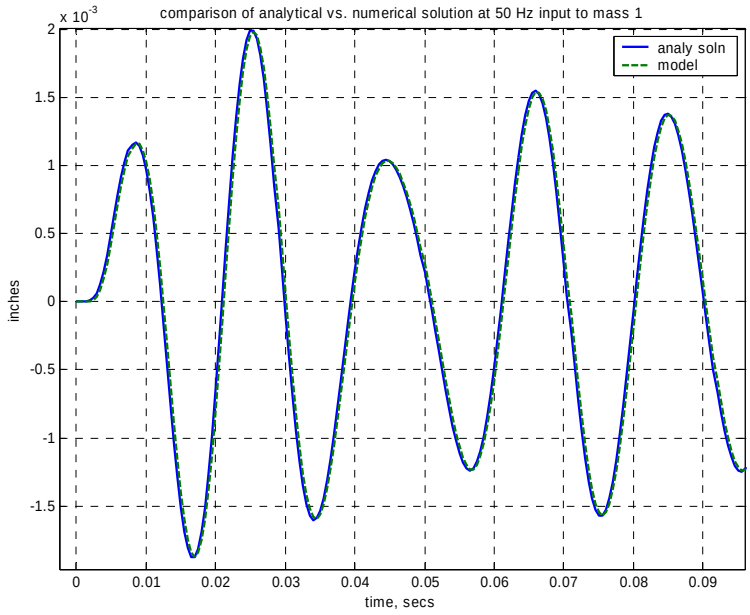


Figure 35 Comparison of Analytical Model to the Numerical Simulation for a 50 Hz forcing frequency

The Bode plot of the model is given below for 0% and 5% damping.

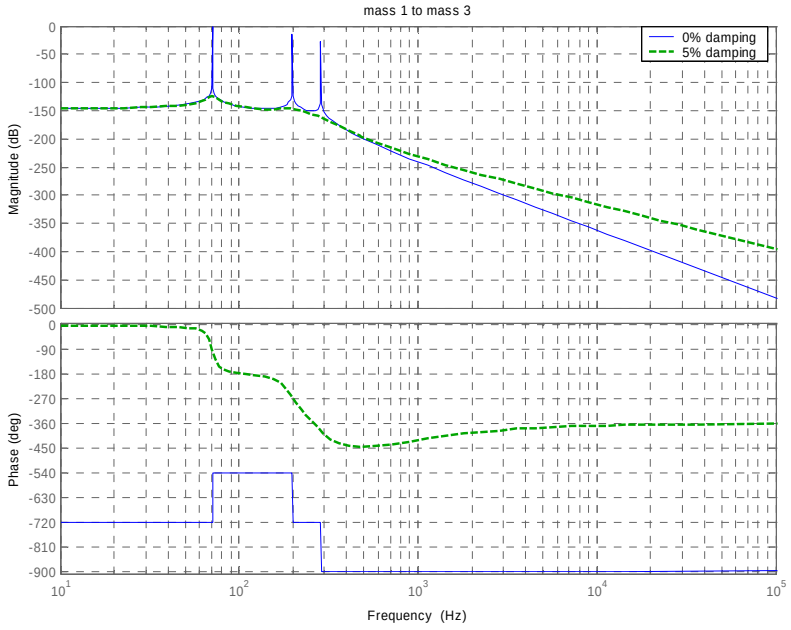


Figure 36 Bode Plot for three mass model, input at mass 1

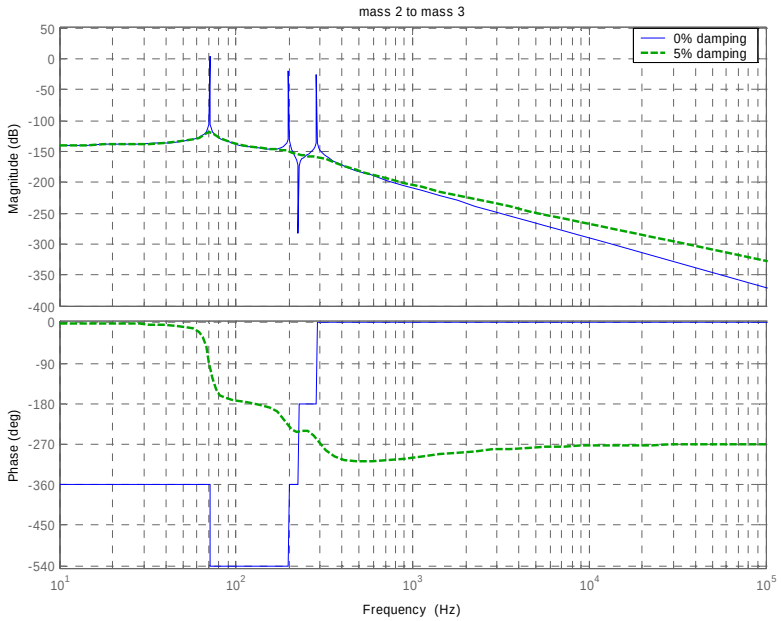


Figure 37 Bode Plot for three mass model, input at mass 2

1. LMS Control of the Three Mass Model

A simple LMS controller is developed using MATLAB/Simulink standard blocks, patterned after the LMS filter provided in the Digital Signal Processing (DSP) Toolbox of Simulink. The LMS filter in the DSP Toolbox does not have the capability to use a "filtered-x" version of the controller. The block diagrams below show the modification.

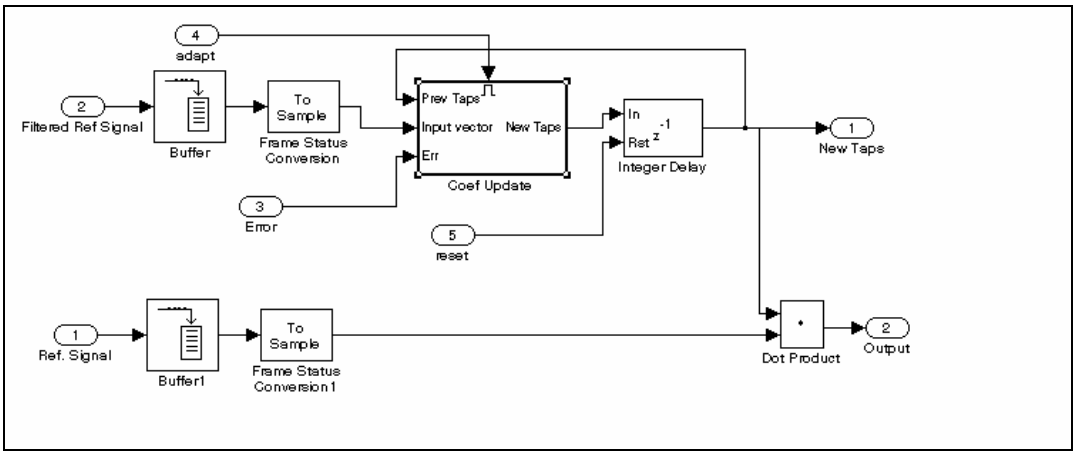


Figure 38 LMS Controller Block Diagram

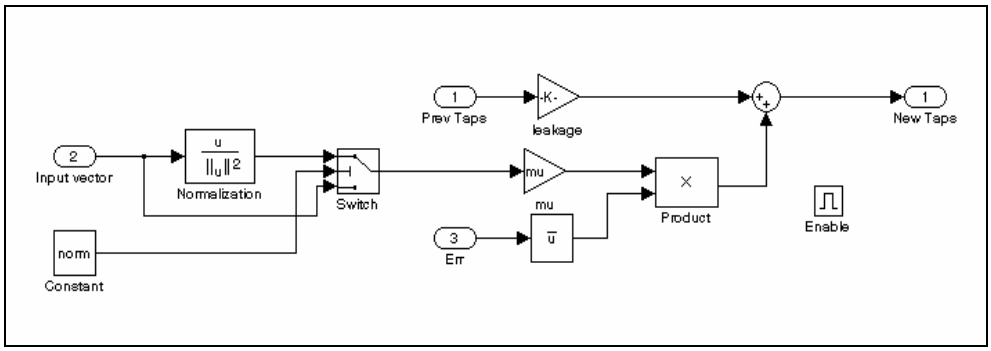


Figure 39 Coefficient Update Block Diagram

The buffers in Figure 38 have length equal to the order of the filter. The overlap for each buffer is set to 1 less than the length, thereby producing a vector of delayed inputs at each time step. The leakage factor in Figure 39 may be used if there is insufficient spectral excitation of the algorithm, such as the case of a noiseless sinusoid [Kuo, 1996, p. 36]. In the studies conducted in this research, the leakage factor was not used and this gain was set to one. The Normalization block was used if the normalized version of the LMS (nLMS) algorithm was desired. This block divides the input vector by the Euclidian-Norm-Squared of the vector. The value "mu" in Figure 39 is the adaptation rate for the LMS algorithm or the value for α in equation (2.27) if the nLMS algorithm is used. Throughout the rest of this study, the nFXLMS algorithm is used, and α is replaced using the symbol μ. The number of stages or order of the filter was determined by running simulations with increasing number of stages to determine the optimum number for the LMS controller. For this problem, we determined that 16 stages is optimum. The reference signal is implemented using a second DSP Sine Wave. The reference signal amplitude and phase may then be varied to study the effects on the controller.

a. Effect of Phase Shift of the Reference Signal

In order to observe the effect of a phase shift of the reference signal, a small adaptation rate was selected. As noted earlier, small adaptation rates result in a smaller value of the misadjustment, and a better steady state response. This can be seen in the figure below. In Figure 40, two 60 second runs were made, one with a large μ and

one with a small value of μ. The disturbance was a 30 Hz signal with a varying bias (described in section *c* below). The plot shows the effect of misadjustment.

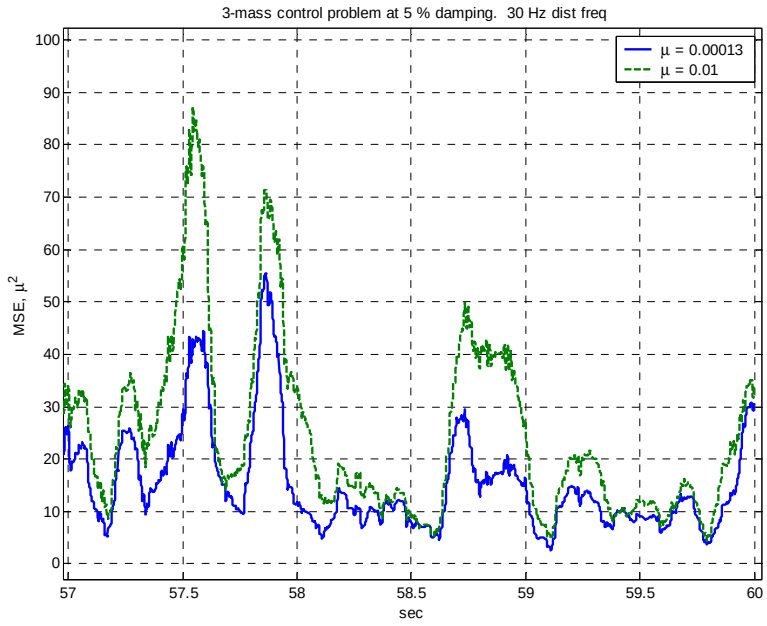


Figure 40 Effect of misadjustment on steady state solution to the three mass system

The adaptation rates selected for the three mass model were on the order of 1 x 10^{-4} for the value of μ in order to observe the effect of phase on the reference signal. Runs were conducted, varying the forcing frequency from 10 to 170 Hz. The amplitude of the disturbance was adjusted to provide approximately 350 microns of motion at mass 3 for each disturbance frequency. The phase of the reference signal was varied from -3.1 to +3.1 radians at each frequency, and the MSE over the last one second of a 10 second run was plotted. The effect of the phase shift is easily seen in the following figures.

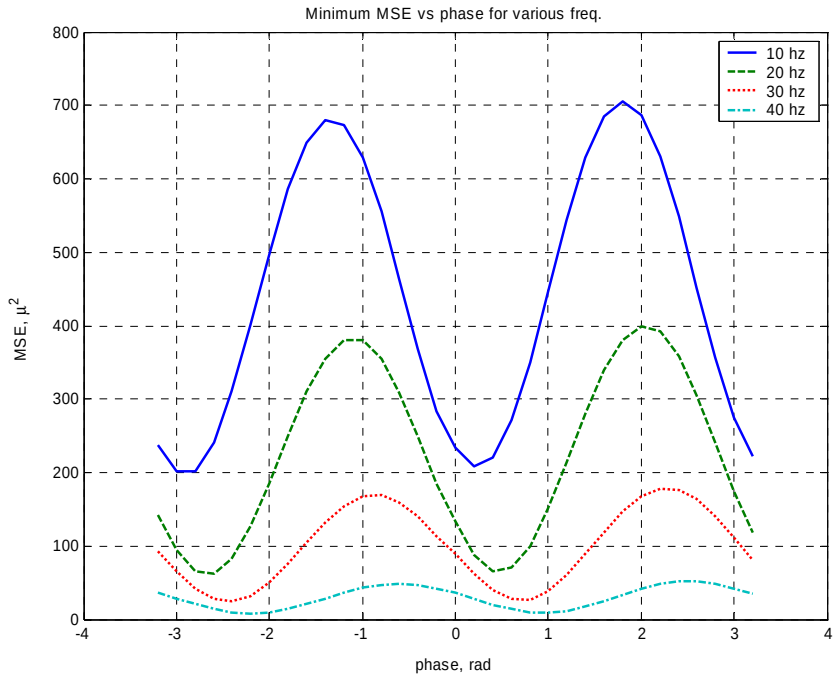


Figure 41 MSE vs phase shift of reference signal for 10 to 40 Hz

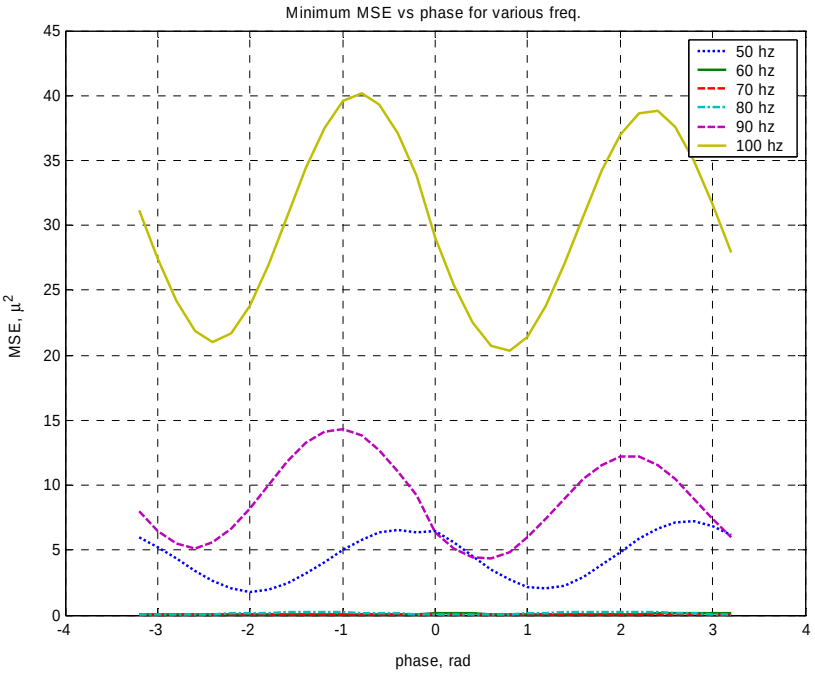


Figure 42 MSE vs phase shift of reference signal for 50 to 100 Hz

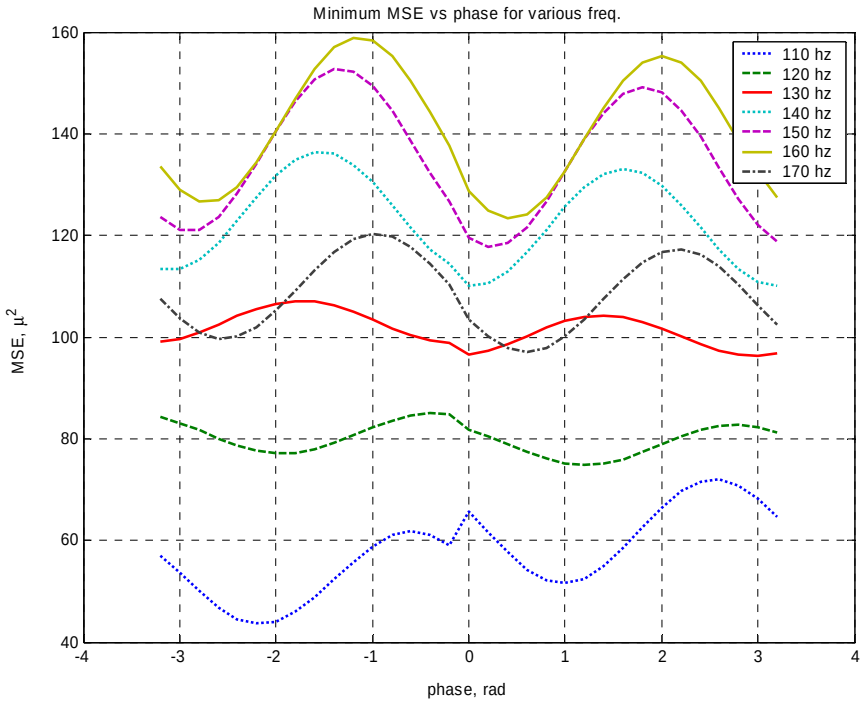


Figure 43 MSE vs phase shift of reference signal for 110 to 170 Hz

The jump in the 110 Hz MSE result in Figure 43 is due to a slight change in the disturbance amplitude that the program caused by the simulation being run in 2 simulations, from -3.1 to 0 radians and then from 0 to +3.1 radians. These figures clearly show that if the optimum phase shift for the reference signal can be used, the convergence is drastically effected for small values of μ.

b. Adaptive Delay Filter

To find this optimum, and use it to increase convergence, an Adaptive Delay Filter (ADF) was developed. The ADF uses variable delays to increase the correlation between the phase of the reference signal and the phase of the error, as discussed in Chapter II. The block diagram of the ADF is shown below.

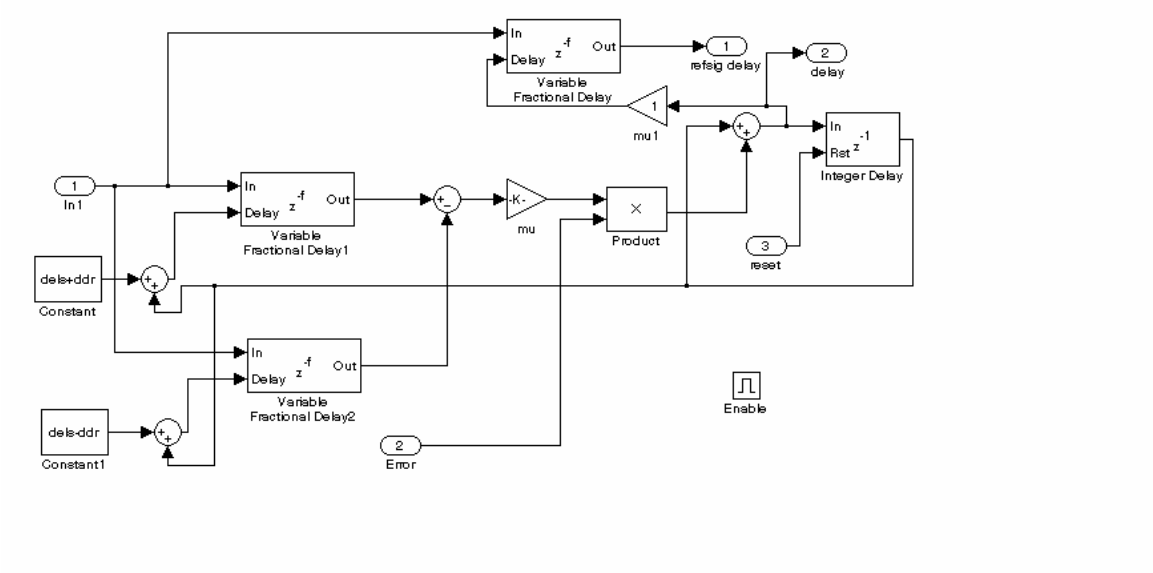


Figure 44 Adaptive Delay Filter

In Figure 44, the reference signal is input at input port 1 (In 1). This signal is then delayed, by the block labeled Variable Fractional Delay and output at output port 1 (refsig delay). The delay is calculated using equation (2.41). The value of μ_Δ used in this diagram is bounded by equation (2.42) and is set to 0.01 for the simulation runs. The constant "dels" is the estimate of the number of samples delay through the secondary plant, and the value "ddr" is normally set to 1 for the calculation involving the symmetric difference (see equation (2.41) and accompanying discussion). Since there is the competing effect between the LMS controller and the ADF of both driving the error to zero, one must give the ADF a good estimate of a starting delay, in order for it to have an effect. Otherwise, the error may be driven to zero before the ADF can adapt. Reviewing the figures above with respect to phase shift, it is noted that a lead angle will give the optimum effect. Therefore, the initial estimate used for the ADF is a delay equal to 90% of one period for the frequency of interest. This gives the ADF a reasonable starting point, and allows the optimum response. As an example of the effect of the ADF, two simulations are run on the three mass system. The first run uses the LMS controller alone, and the second, the LMS and ADF controller.

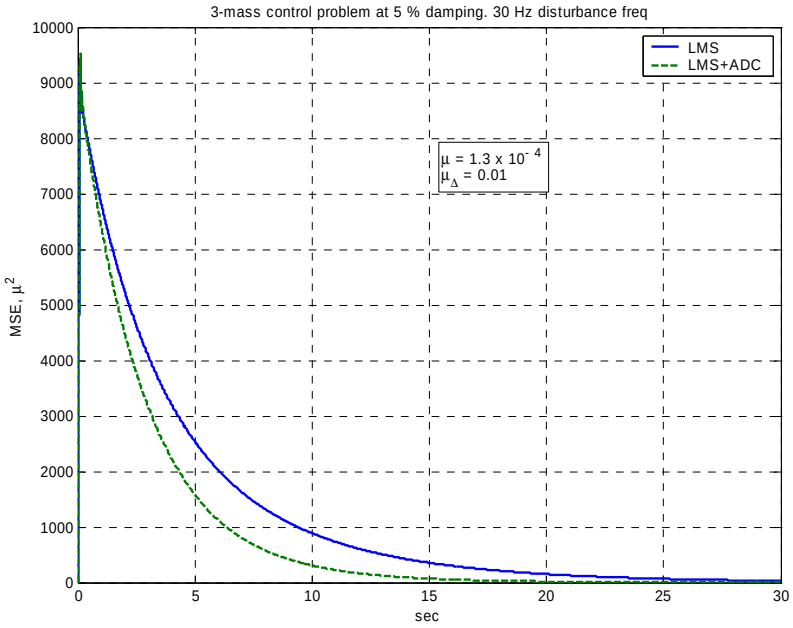


Figure 45 Comparison of LMS control with LMS+ADF

Figure 45 clearly shows that use of the ADF improves convergence speed for this type of control problem. Misadjustment has not been affected.

c. Effect of Bias on the Control System

In using adaptive control to remove unwanted signals, such as in an electronic signal, the bias in the signal is normally removed prior to filtering. In the control of a laser beam with "noise", there is usually a bias associated with the signal due to the fact that the beam does not hit the center of the target. Generally, this bias is removed by means of a compensator, and the adaptive filter/controller removes the noise. We noted during testing on the LJC testbed, that a small amount of bias added to the reference signal contributed significantly to the convergence of the algorithm. This effect was discussed in Chapter II, and is examined here using the three mass model. In observing the bias on the LJC testbed, a certain amount of randomness is evident. The figure below illustrates the laser beams x coordinate on the target sensor over 1.5 seconds with no movement of control mirrors:

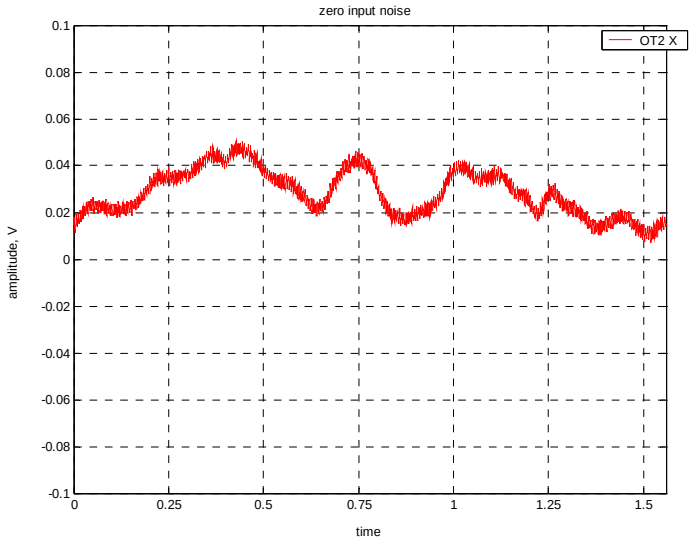


Figure 46 Bias of Laser Beam on Target

This variance is due to the effect of thermal gradients in the air surrounding the optical bench and platform, and due to the variance in the laser source. In order to accurately examine the bias effect on the three mass model, a similar varying bias was constructed using random noise and an averaging block in MATLAB.

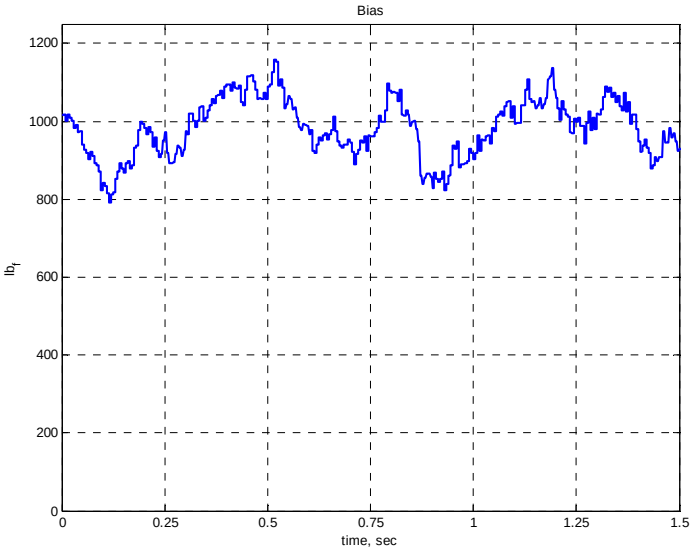


Figure 47 Variable Bias simulated in MATLAB

The Adaptive Bias Filter (ABF) discussed in Chapter II is used to add bias to the reference signal. The figures below illustrate control with and without the ABF, for small values of μ on the order of 1 x 10^{-4}. The ABF was initialized using an estimate of *C*' in equation (2.46) of 70% of the value of the mean bias present. Figure 48 shows the effect of the added bias. The position of mass three is centered at a value different from zero, corresponding to the bias in the disturbance signal. In Figure 49 this bias is removed. The Mean Square Error plot in Figure 50 clearly shows the advantage of adding bias to the reference signal.

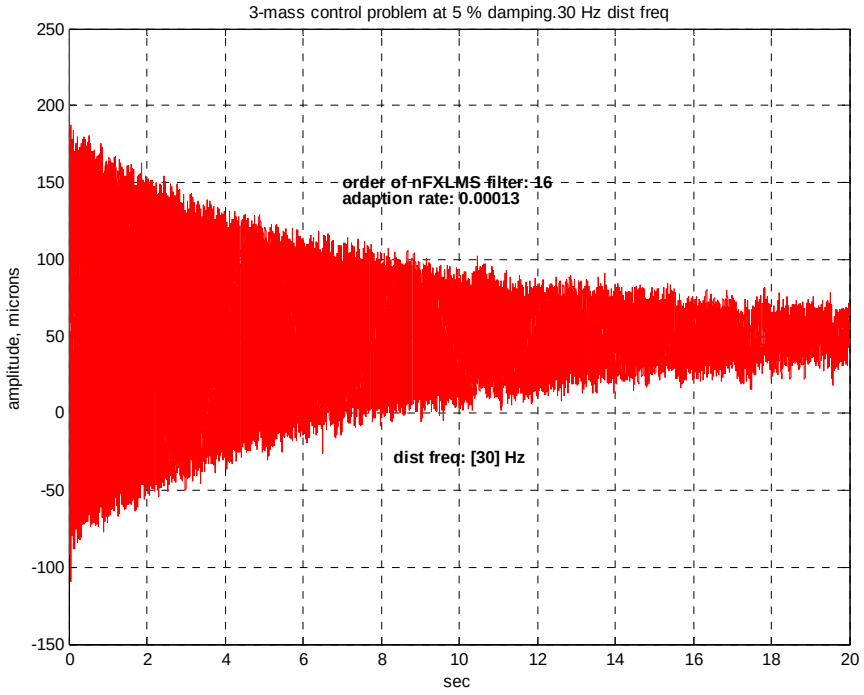


Figure 48 Time Plot – 30 Hz disturbance control using LMS

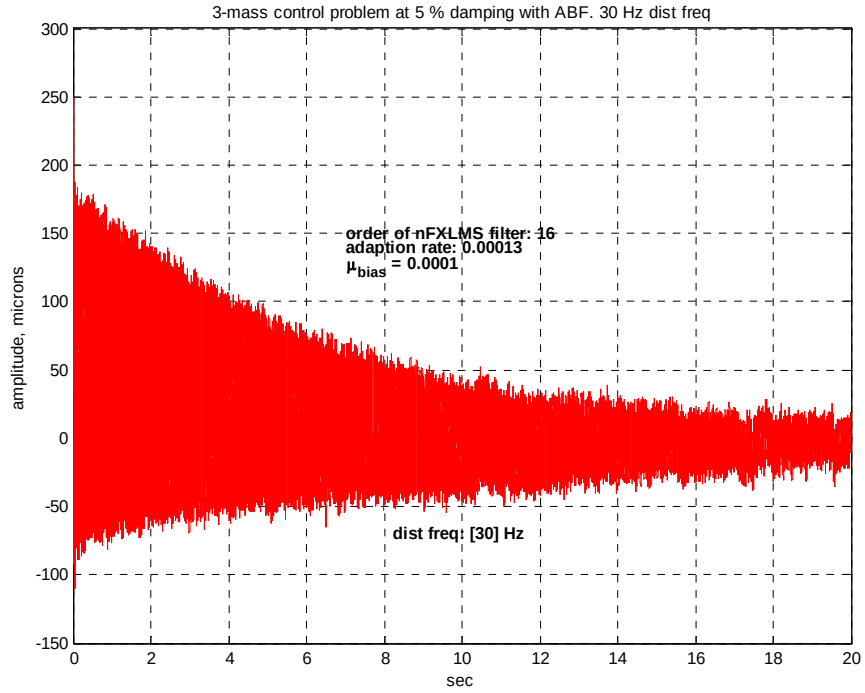


Figure 49 Time Plot – 30 Hz disturbance control using LMS+ABF

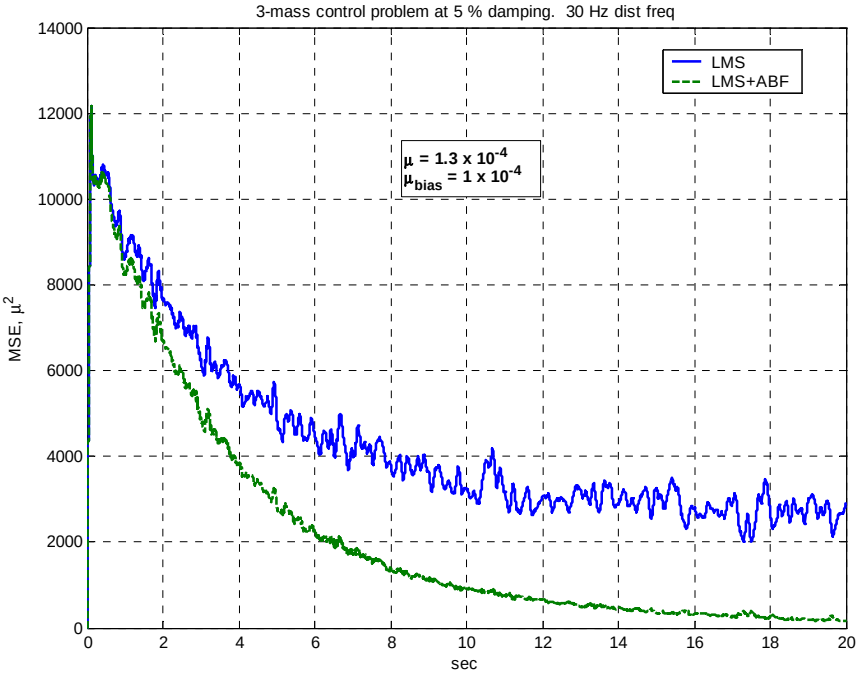


Figure 50 MSE Plot – 30 Hz disturbance with varying bias

Unlike the ADF, the ABF does have an effect even at high values of μ. Without any bias in the reference signal, the LMS controller will not "zero" the signal, resulting in a constant MSE, much higher than an LMS controller with bias added to the reference. This can be seen in the MSE plot for two simulations, one without bias and one with the ABF adding bias.

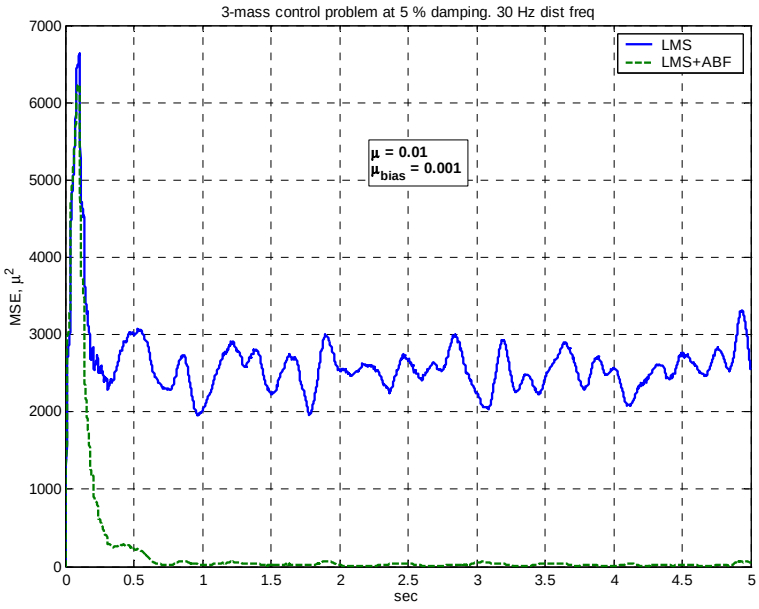


Figure 51 Comparison of LMS and LMS+ABF at high values of adaptation

d. Effect of Anti-resonance Point on the Control System

In Figure 37 an anti-resonance point can be seen at about 225 Hz. Control at this frequency would be expected to be not as effective as at other frequencies due to the loss of effectiveness of the motion at mass 2 affecting the motion at mass 3. This is borne out in Figure 52. At the anti-resonance point at 0.1% and to some extent 1% damping, control is not as effective as at the frequencies on either side of this point. This effect diminishes as the damping is increased. The effect is similar to that which is seen when an actuator is placed near a node for a prevalent mode shape in a structure. As the platform and control system did not exhibit an anti-resonance point, we were unable to verify this experimentally.

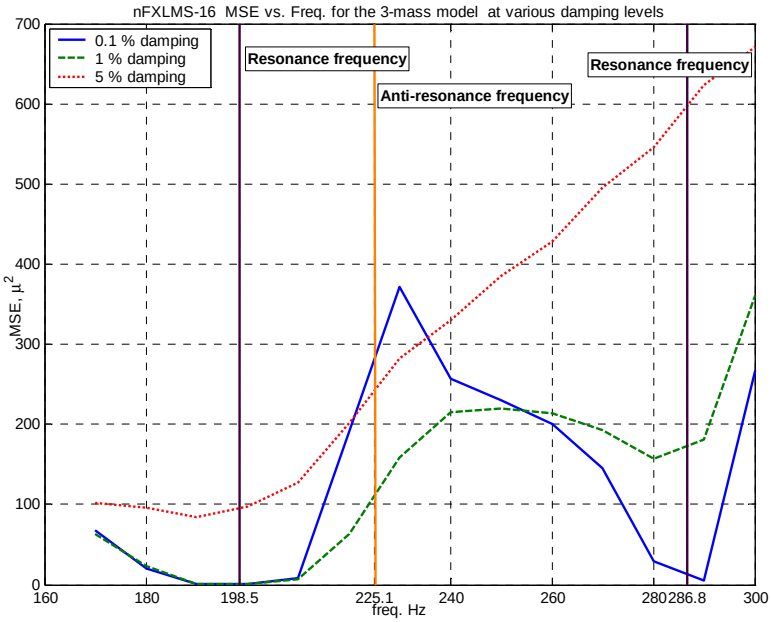


Figure 52 Effect of Anti-resonance point on LMS control

2. Gradient Adaptive Lattice (GAL) Control of the Three Mass Model

Another form of adaptive control is that which uses an adaptive lattice as discussed in Chapter II. The algorithm developed by Haykin [Haykin, 2002, p. 542] was modified for use with MATLAB/Simulink. The algorithm could not be implemented using standard Simulink blocks and was therefore coded for use with an S function. The C code for the S function is provided in Appendix D. Again, the optimum number of stages was determined in the same manner as the LMS filter discussed in section V.C.1 above. 16 stages were used in the following numerical simulations. The reference signal was also provided by a separate DSP Sine Wave source block, in order to adjust bias, phase and amplitude of the signal.

The GAL is similar to LMS control in that it is affected by phase differences and bias. However, GAL is much more sensitive than the LMS controller in these areas. The GAL algorithm uses an adaptation coefficient in a manner similar to the normalized LMS algorithm in that it varies with time, but instead of being normalized by the power of the reference signal, the GAL adaptation rate is normalized by the "power" in the prediction error. Thus, similar to the LMS we can evaluate slow and fast adaptation rates. We

cannot, however, compare values of adaptation rates between controllers, due to the difference in the normalization signals. Unlike the LMS algorithm, the effect on the misadjustment is slight for this type control problem, as can be seen in the following plot:

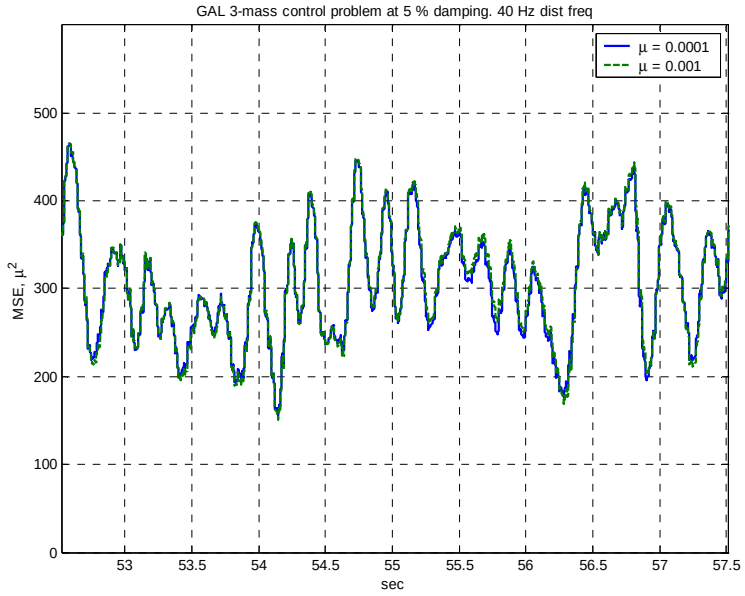


Figure 53 Misadjustment in GAL controller

The speed of convergence is rapid for higher values of μ as can be seen below:

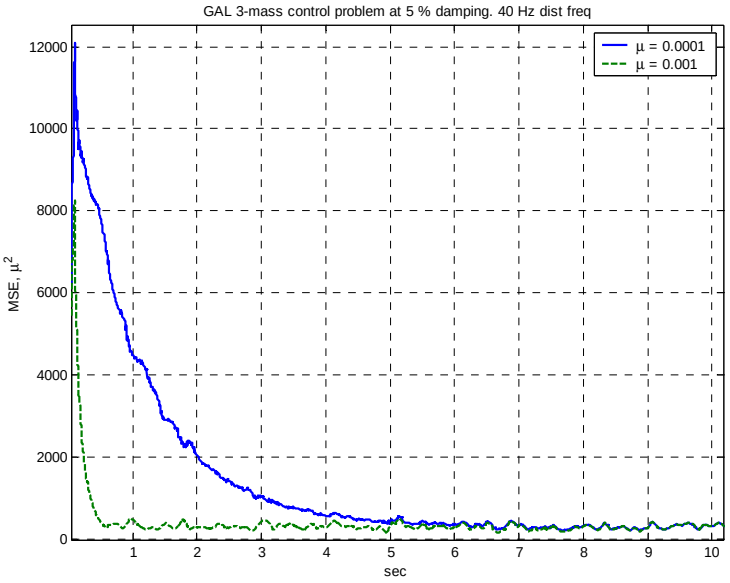


Figure 54 Speed of convergence for the GAL

a. Effect of Phase Shift of Reference Signal

With the speed of convergence so fast, and the misadjustment slight for large values of μ for this problem, correlating the reference and error signals does not contribute appreciably to the solution of the control problem. However, there *is* an effect, similar to the LMS controller, for low values of μ. Therefore, for problems with large eigenvalues of the input correlation matrix **R**, a low value of μ may be necessary and the use of a phase shift would aid in convergence (see equation (2.28)). We note that the amount of phase shift to produce an effect on the GAL convergence is far less than that necessary for the LMS convergence. The GAL algorithm is much more sensitive to the shift than the LMS, and if the phase is shifted too far, the algorithm becomes unstable.

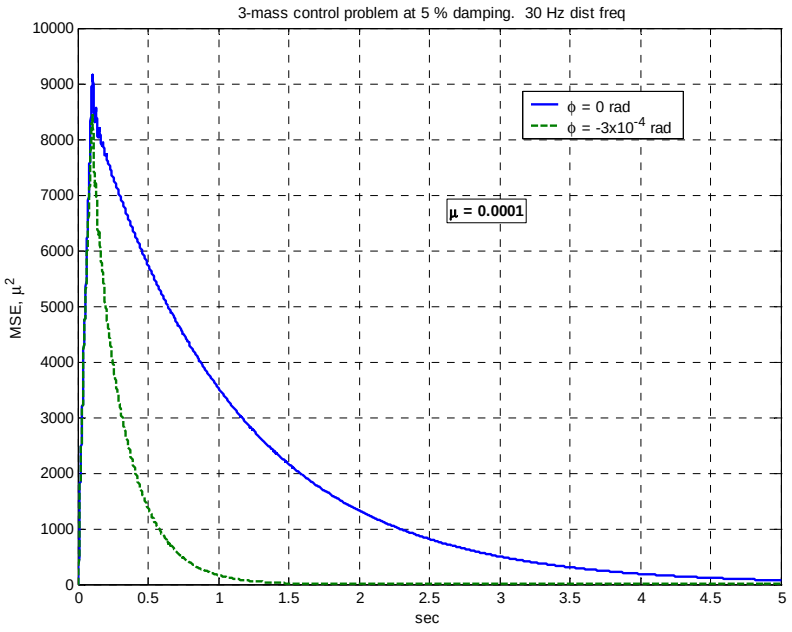


Figure 55 Effect of Phase shift of reference signal on GAL with low values of μ

b. Effect of Bias on the GAL Control System

Again, similar to the effect of bias on the LMS system, the GAL is affected by bias as well. A varying bias is applied to the disturbance signal as in the LMS problem. Bias is then added to the reference signal. Two effects are noted. First, the addition of bias to the reference signal allows the algorithm to converge faster, as well as center the third mass' position at zero. Second, without bias added to the reference

signal, any increase in the value of μ results in an unstable controller. With bias added, μ may be increased by an order of magnitude, resulting in rapid convergence *and* centering mass 3 at zero.

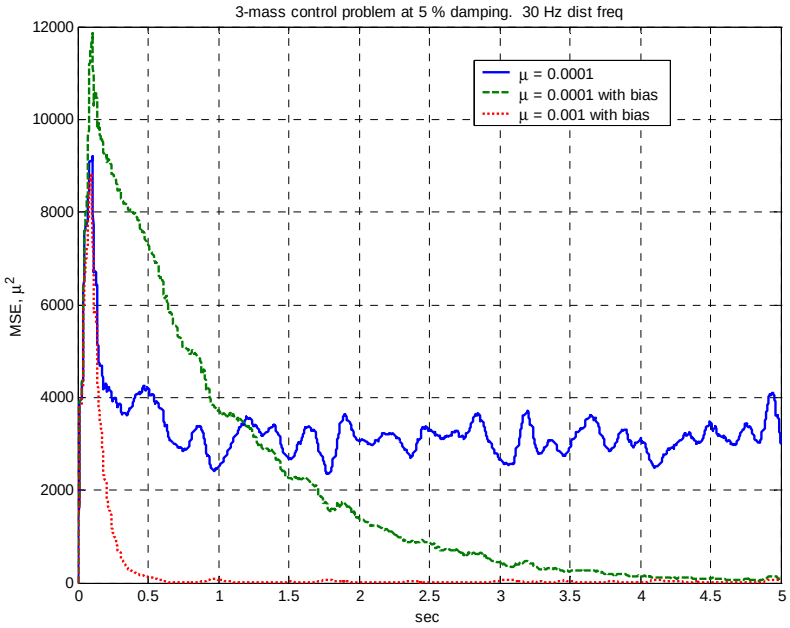


Figure 56 Effect of bias on the GAL controller

3. Comparison of GAL and LMS

Our goal is to rapidly correct the bias and narrow band frequency present in the disturbance. Our method of comparison will be the Mean Square Error (MSE). Our measure of effectiveness will be the convergence time, and the misadjustment. The misadjustment will be compared by observing the MSE for each controller at steady state. In the following simulations, each run will have a varying bias, as in Figure 47. We will start with the 30 Hz disturbance previously discussed, but then evaluate the controllers at frequencies above the first fundamental, as well as multiple frequencies. Both the LMS and GAL controllers will use 16^{th} order filters.

a. Rate of Convergence and Misadjustment for a Single Frequency

In order to compare the two controllers, the adaptation rate will be increased until the point just prior to instability. This will give us the fastest convergence for each controller. We will then compare the convergence rate and misadjustment.

Steady state will be determined by stability of the tap gains. Due to the emphasis on convergence at high values of μ, the ADF will not be used for the LMS controller. In these comparisons, the GAL controller will have the optimum bias added to the reference signal, and the LMS controller will use the ABF.

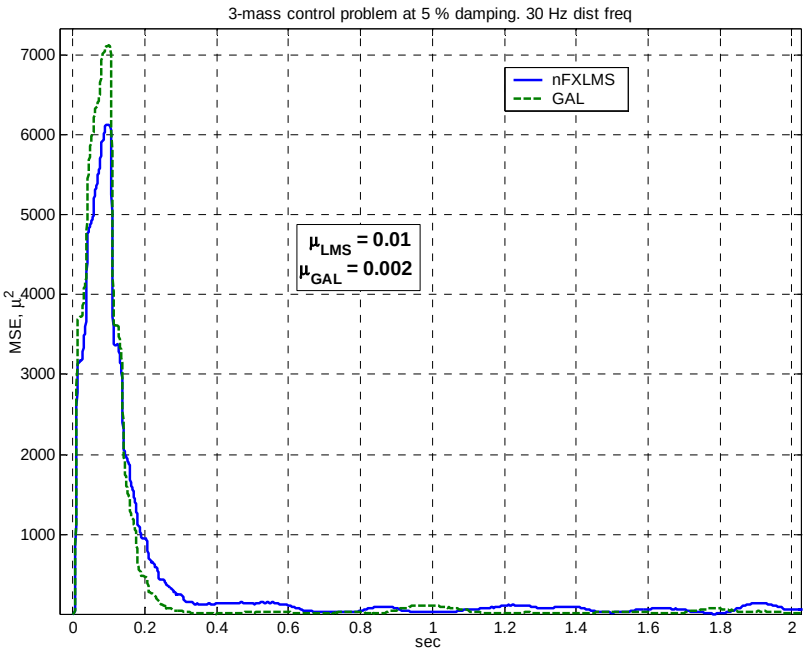


Figure 57 Comparison of MSE for GAL and LMS, 3-mass system at 30 Hz

As can be seen in the figure above, the GAL controller has a slightly faster convergence. The misadjustment is very nearly the same, with the mean value of the MSE for the LMS controller over the last 10 seconds of a 20 second run being 0.5 micron^2 larger than the GAL controller. A comparison of the steady state response is shown in Figure 58. A comparison of the Power Spectral Density (PSD) plot is shown in Figure 59. The PSD is calculated over the last 5 seconds of the 20 second simulation. From the PSD, we see that the LMS controller removes more of the 30 Hz disturbance than the GAL, but excites the first mode of the system as can be seen by the energy added at 70 Hz.

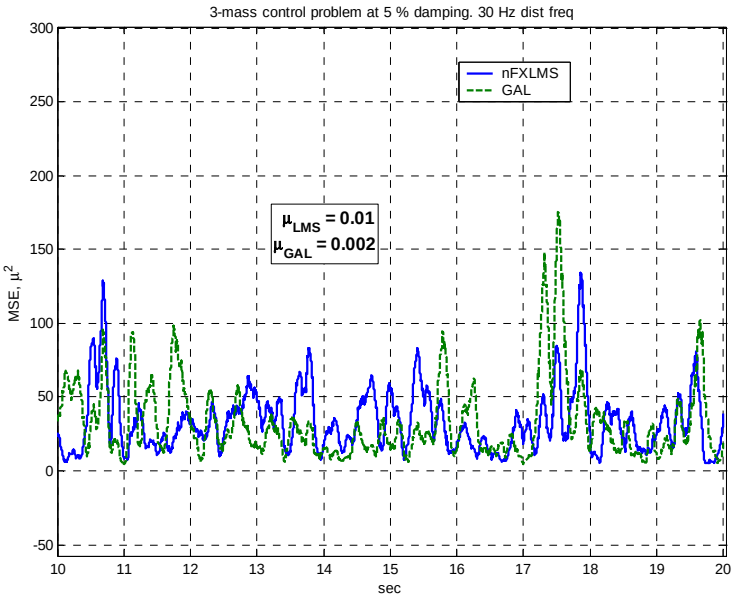


Figure 58 Comparison of Steady State for LMS, GAL, 30 Hz disturbance

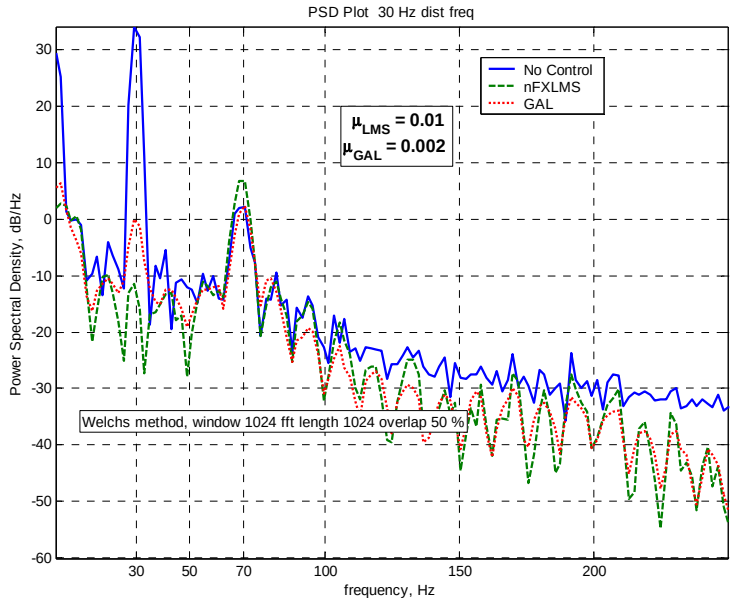


Figure 59 Comparison of controller's PSD response for 30 Hz

Now we will compare controllers for a single frequency above the first fundamental of 70 Hz. The same simulation with varying bias will be run again at 87 Hz. An interesting phenomenon occurs above 70 Hz (the first phase shift between input and

output of the 3-mass system) for the GAL controller. It becomes *unstable*. In order to stabilize the output of the controller, the output of the GAL filter must be inverted and the adaptation rate significantly decreased. If the GAL controller is used to remove an 87 Hz electronic signal, (i.e. the disturbance does not go through a transfer function), it works as expected and no inversion is required. It is only when there is a transfer function between the disturbance and the output that provides a phase shift that the instability occurs. As discussed above, the GAL controller is very sensitive to phase.

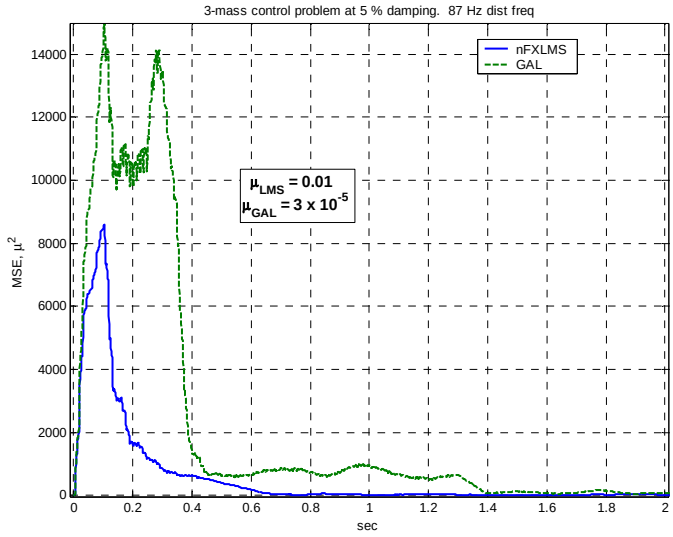


Figure 60 Comparison of MSE for GAL and LMS , 3-mass system at 87 Hz

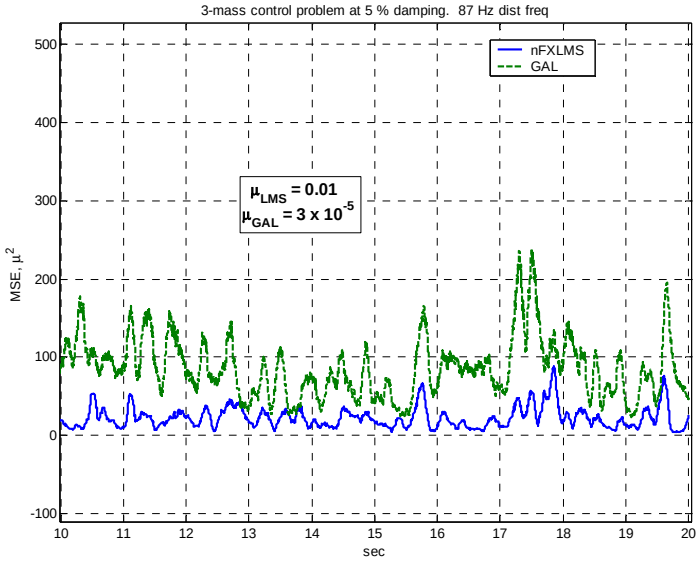


Figure 61 Comparison of Steady State for LMS, GAL, 87 Hz disturbance

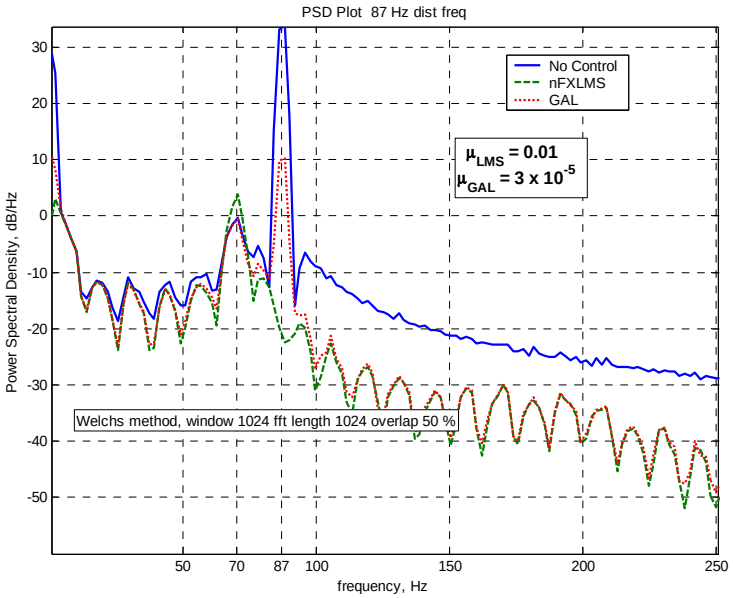


Figure 62 Comparison of controller's PSD response for 87 Hz

As can be seen from the above figures, the LMS controller converges faster and has lower misadjustment than the GAL controller at these frequencies. The LMS controller removes the narrowband disturbance, but still adds energy to the first mode, as in the 30 Hz case.

b. Rate of Convergence and Misadjustment for Multiple Frequencies

Our first comparison will be for two frequencies below the first fundamental. Both narrowband components have the same amplitude. Figure 63 shows that the LMS controller actually converges faster than the GAL. Additionally, the LMS controller has a better steady state response than the GAL. In order to minimize the instabilities in the GAL controller, low values of μ must be used, which leads to slower convergence.

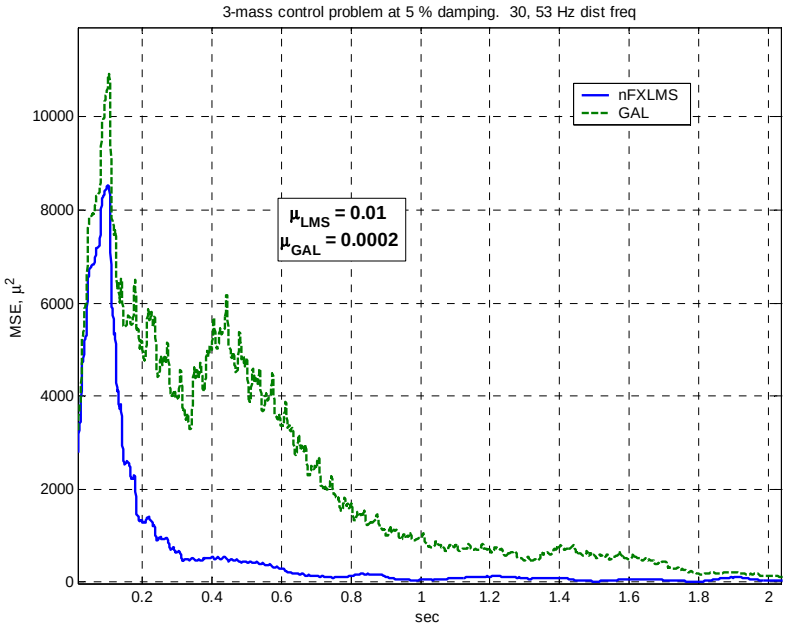


Figure 63 Comparison of MSE for GAL and LMS ,3-mass system at 30 and 53 Hz

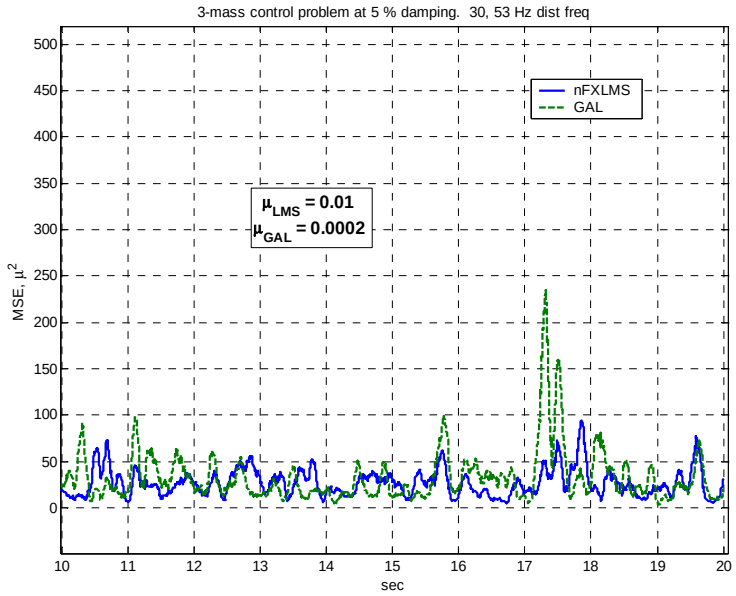


Figure 64 Comparison of Steady State for LMS, GAL, 30 and 53 Hz disturbance

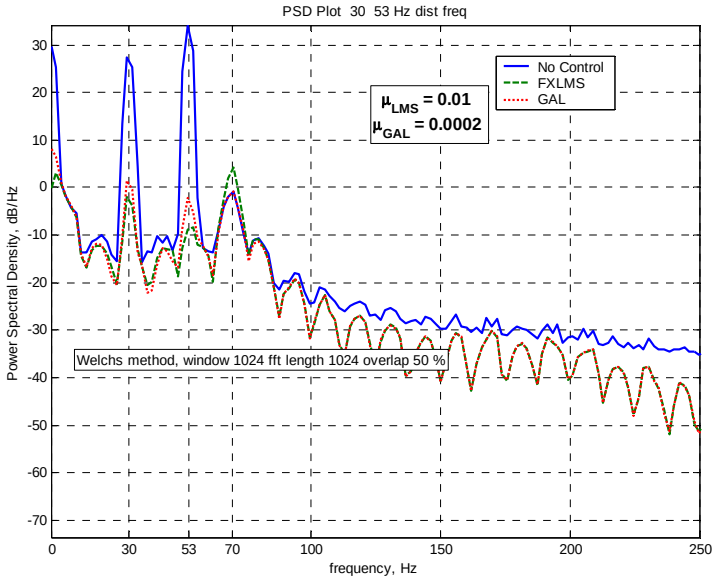


Figure 65 Comparison of controller's PSD response for 30 and 53 Hz

Our second simulation of multiple frequencies involves one narrowband frequency on either side of the first fundamental. In this case, the GAL is unstable except at very low adaptation rates, or when an IGRS is used.

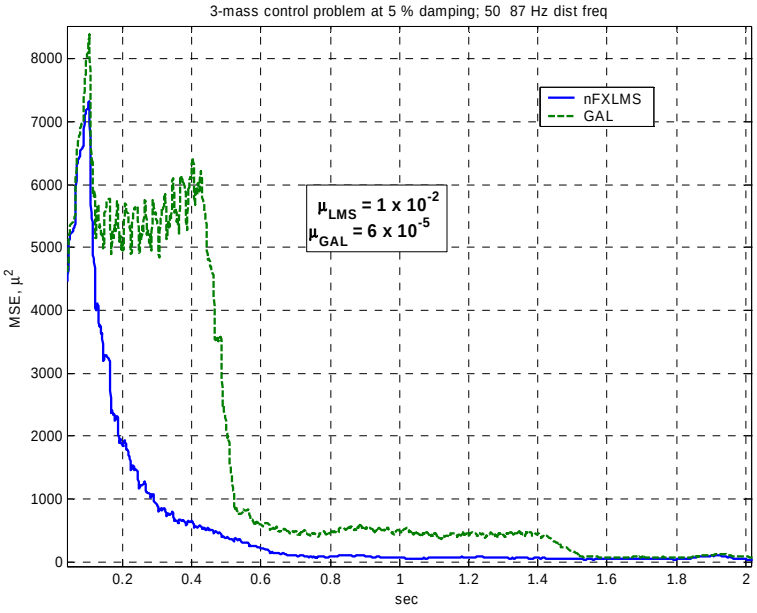


Figure 66 Comparison of MSE for GAL and LMS ,3-mass system at 50 and 87 Hz

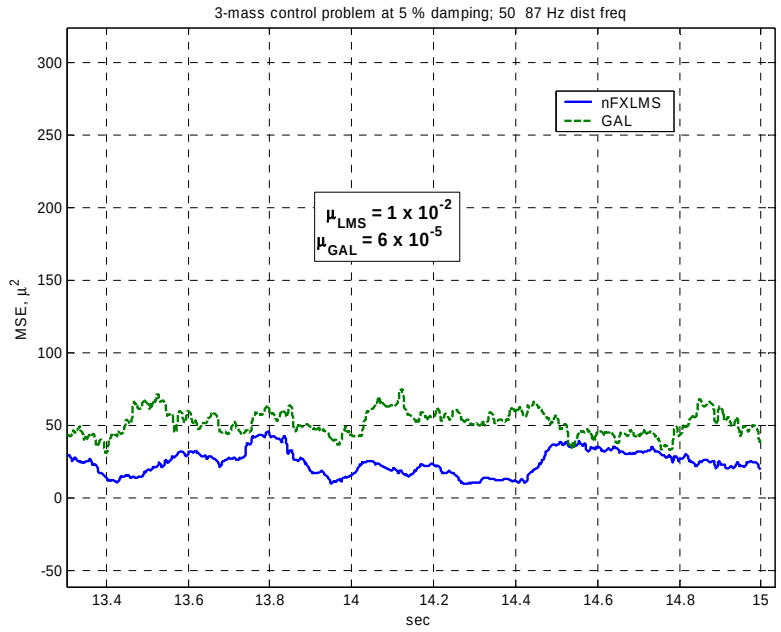


Figure 67 Comparison of Steady State for LMS, GAL, 50 and 87 Hz disturbance

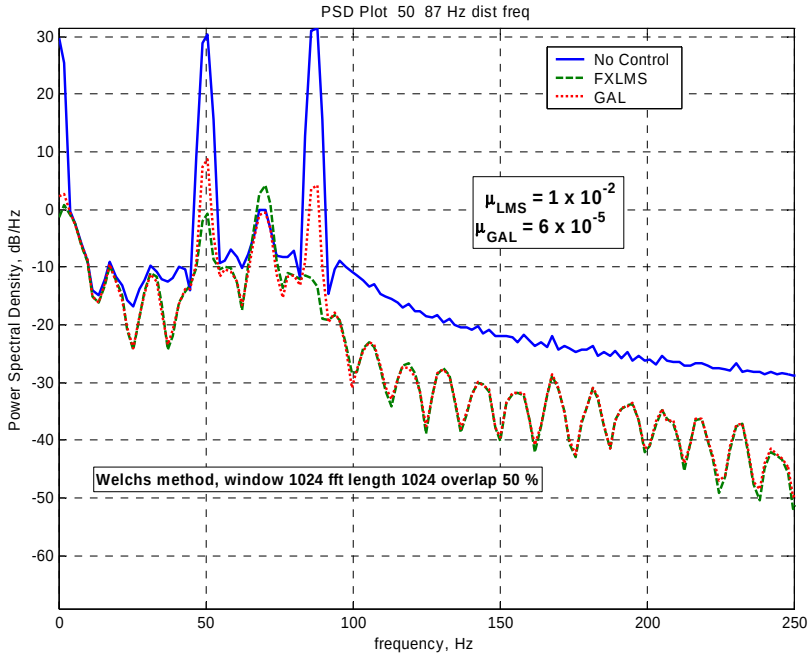


Figure 68 Comparison of controller's PSD response for 50 and 87 Hz

As can be seen in Figure 66 through Figure 68, the GAL controller is not effective in controlling multiple frequencies, when one frequency is above the first phase

shift. Therefore, in systems in which multiple modes are excited, if some of the natural frequencies are on either side of a phase shift in the frequency response function, the GAL will have trouble controlling the system, depending on the relative amplitude of the narrowband peaks.

c. The Addition of Random Noise to the Forcing Function

Since we are also interested in the effect of random noise in the signal for the LJC testbed, random noise was inputted into the forcing function for the 3-mass model. The control of 200 Hz band-limited white noise in combination with two narrowband frequencies, 50 and 87 Hz as before, was simulated. For this case, we remove the variable bias in order to just examine the effect of random noise in the forcing function. Additionally, we use an IGRS for the GAL controller to correct the phase difference that occurs for the 87 Hz frequency.

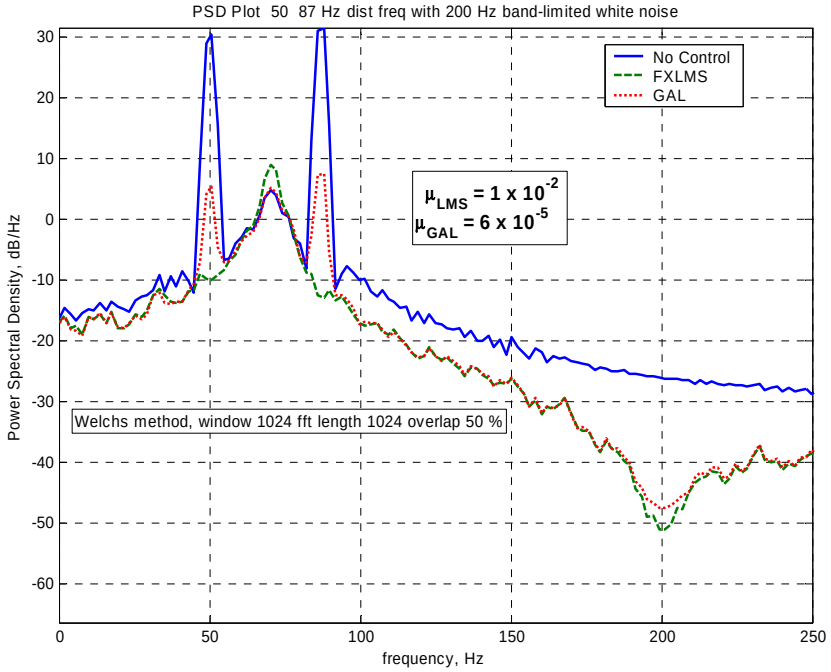


Figure 69 Comparison of controllers PSD response for multiple frequencies and white noise

Once again, the GAL is not as effective in controlling the two frequencies as the nFXLMS controller.

D. SUMMARY

Simulation of the control problem using the 3-mass model allowed us to evaluate the effect of bias and phase shift on the reference signal. The simulation showed that a phase shift of the reference signal could increase the rate of convergence for an adaptive filter, without increasing the misadjustment. The addition of bias to the reference signal could correct a biased error signal, even one that is slowly varying. The simulation also shows that the GAL filter is very sensitive to the difference in phase between the reference signal and the error, resulting in instabilities unless the phase difference is corrected by means of an IGRS.

THIS PAGE INTENTIONALLY LEFT BLANK

VI. DISTURBANCE REJECTION EXPERIMENTS ON THE LJC TESTBED

The disturbance rejection experiments on the testbed were performed in three phases. The first phase involved the control of the laser beam with the platform stationary. This allowed us to evaluate the performance of the control system under ideal conditions, as well as verify the operation and setup of the testbed. The second phase involved shaking the platform with discrete frequencies, while maintaining the DFSM at a constant position. This allowed us to see the effect of shaking the control system, and observe the control of vibratory motion by the RFSM. The third phase consisted of a randomly disturbed laser beam by the DFSM, and vibration of the platform. This would be what we would consider the on-orbit or on-station model. Our measure of effectiveness will be once again the Mean Square Error (MSE), determined by a running average filter of window 500 of the squared error. We will use the value of one standard deviation of the laser beam's position about its mean to evaluate the "tightness" of the beam – how far it has spread out. We also measure the mean value of the beam's position to determine how close to the center of the target the controller is placing the beam.

A. LQR CONTROLLER FOR THE TESTBED

In order to compare the classical control systems with the adaptive, an LQR controller was constructed using equations (4.1) and (4.2) as the state space system of equations. The linear quadratic optimal gains are determined by using the command "LQRD" in MATLAB with a sample rate of 2 kHz, which calculates the required gains for the discrete system. The weighting matrix (**Q**)for the states of the cost function (see equation (2.1)) is an identity matrix, except for the values corresponding to the states for the voltage at the detector which are given a weight of 1000. The input weighting matrix (**R**) is given a value of 0.1. The Linear Quadratic Regulator requires the measurement of each state in the state space system. In the testbed, we measure the voltage at the detector and the position of the RFSM, but not the velocity of the mirror in the RFSM, which is one of the states of the system. A Kalman estimator was constructed to estimate

the velocity of the mirror. The plant model in this state space system does not have an integrator. For the LQR to center the beam on target, an integrator was added to the model after the technique described in Ogata [Ogata, 2002, pp. 847-850].

The linear quadratic optimal gains calculated in this manner were then adjusted to obtain the tightest controlled beam possible. The program used to calculate the state-space system and the resulting output are provided in Appendix E.

B. STATIONARY PLATFORM EXPERIMENTS

Figure 8 is reprinted here for ease of reference.

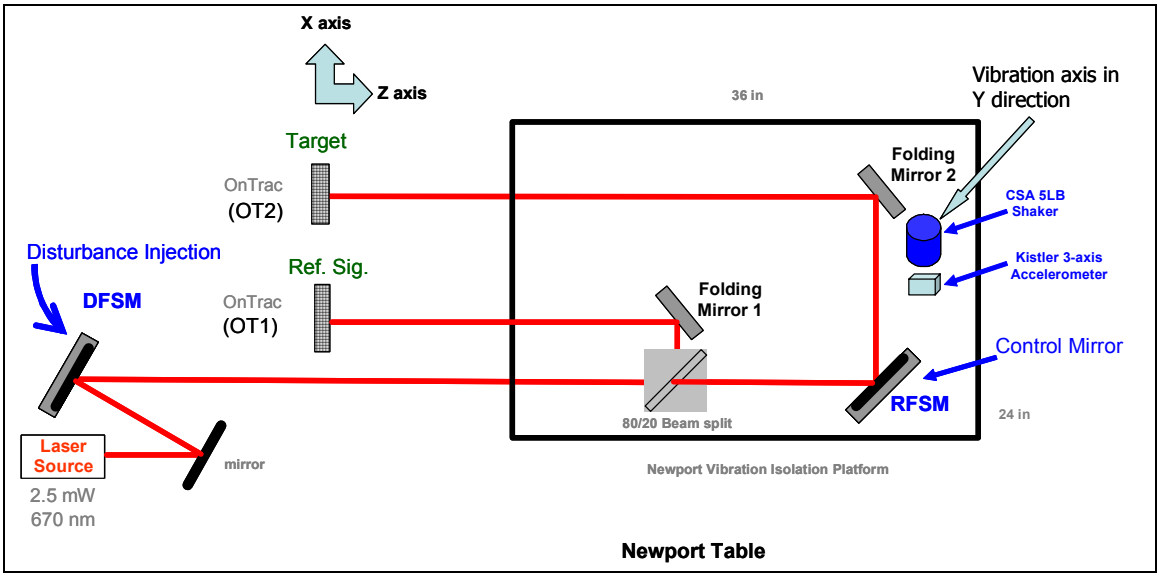


Figure 70 Laser Jitter Control Testbed

In the stationary platform experiments, the vibration isolation platform and the Newport Table are floated. The DFSM is used to inject a disturbance (narrowband, broadband or a combination) in the laser beam from the source. The Position Sensing Detector (PSD) labeled OT1 is used as a reference signal or as a signal to synchronize the Internally Generated Reference Signal (IGRS). The PSD labeled OT2 is the target for the beam. The power supply for the shaker is secured. The output from the PSDs are labeled X and Y axis for each PSD. The same controllers used for the 3-mass model are used for the testbed. The number of stages or order of the filter was determined by running experiments with increasing number of stages to determine the optimum number for each

of the controllers. For the LJC testbed, 24 stages was optimum for the nFXLMS controller and 18 stages for the GAL controller. The "filter" for the nFXLMS controller is the state-space system of equations (4.1) and (4.2). The bias added to the IGRS for the nFXLMS controller is the mean value of the bias present at the OT1 sensor, and though bias is in the reference signal, it is not optimized.

1. Periodic Disturbances- Phase Effect

In the same manner as for the 3-mass model, the IGRS was phase shifted to determine the effect on the convergence of the nFXLMS controller for low adaptation rates. The experiment was run for 5 seconds, with the controller starting at 1.6 seconds into the run. This allowed time to record the disturbance at the target prior to control, in order to make comparisons before and after the controller was running. The MSE was calculated for the last 1 second of the run and used for comparison. The figures below provide the results for the Y axis of the PSD. The X axis is similar. The adaptation rate for the nFXLMS algorithm was 0.15 for the X-axis of the FSM.

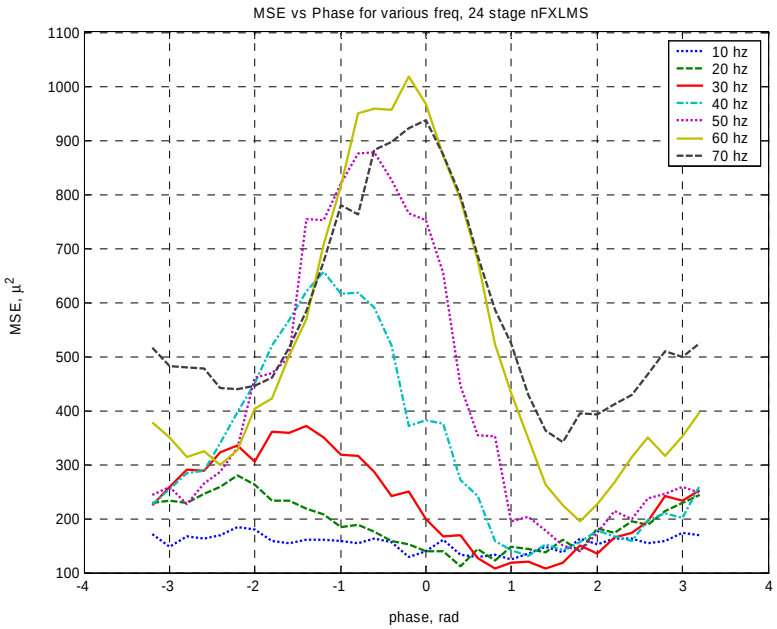


Figure 71 Effect of Phase shift of the IGRS for the testbed, low frequency

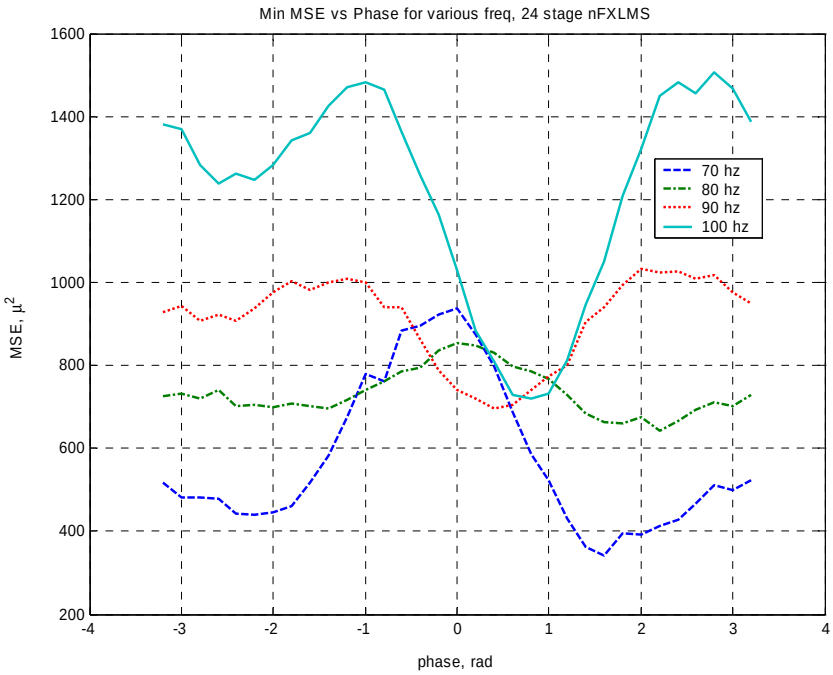


Figure 72 Effect of Phase shift of the IGRS for the testbed, high frequency

As can be clearly seen, the phase shift has an effect on the convergence. In fact, for 50 through 70 Hz, it has a dramatic impact, improving the response by a factor of 5 over the non-phase shifted IGRS for the 60 Hz disturbance.

The GAL was also tested for the stationary platform case with periodic disturbance. The GAL controller uses the reference signal provided by OT1. The following figure compares the MSE for the GAL controller with the MSE for the nFXLMS controller, reference signal from OT1, and with the MSE for the nFXLMS controller using an IGRS and the optimum phase shift determined from Figure 71 and Figure 72. The MSE was determined from the last one second of a 4.4 second run.

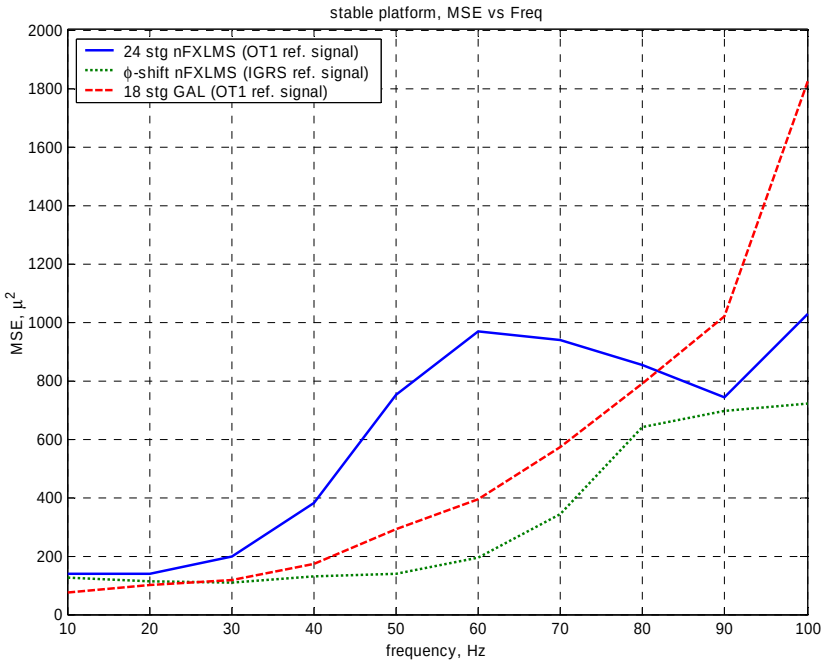


Figure 73 MSE vs. Frequency for nFXLMS, phase-shifted nFXLMS, and GAL controllers, Stationary Platform

Figure 73 shows that the GAL controller develops a lower MSE at 4.4 seconds than the nFXLMS controller using the same reference signal, for frequencies below 80 Hz. However, the phase-shifted nFXLMS develops even lower MSE than the GAL above 30 Hz.

In order to compare convergence rates, a single 50 Hz disturbance was injected by the DFSM for one axis, and the time vs. MSE plotted. The IGRS was modified by the ADF shown in Figure 44 for one of the runs. The ADF was initialized with a phase shift equal to 90% of the period of the disturbance signal. Additionally, a plot of the effect of using an IGRS with no phase shift is included for comparison. The figure shows that the GAL does reach its final value faster than the LMS filters, as is to be expected for this single frequency, however the GAL steady state value is higher than the LMS. The nFXLMS controller with ADF provides a 40% improvement in convergence rate over the nFXLMS using the IGRS alone, and a 50% improvement over the nFXLMS using OT1 as the reference signal.

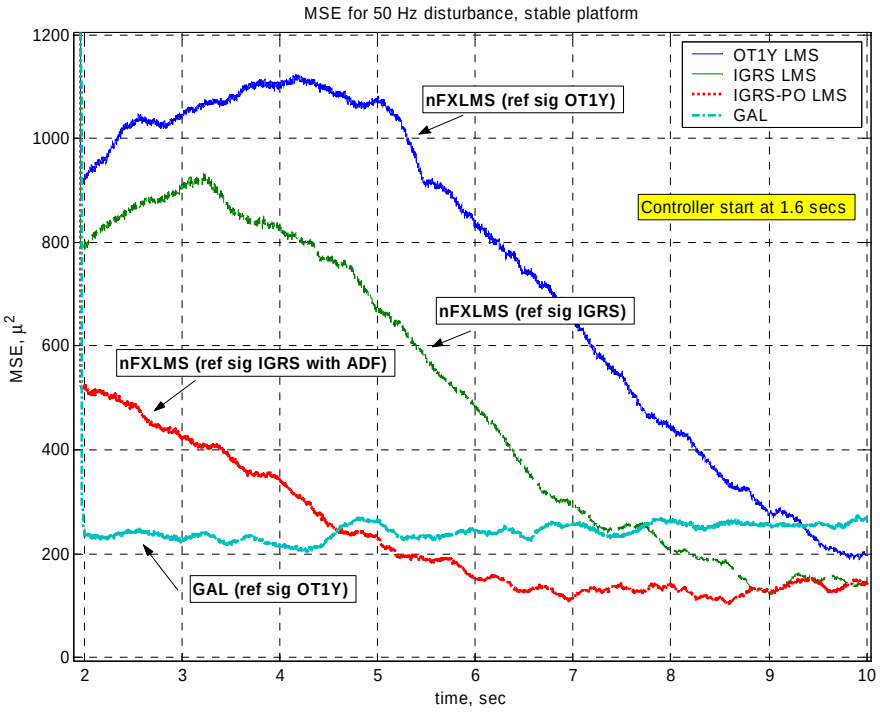


Figure 74 Comparison of controllers for stationary platform

2. The Effect of Bias on the Reference Signal for the Testbed

During our first attempts to use an IGRS with the nFXLMS algorithm, we noted that the controller was far worse (in some cases adding energy to the disturbance) using the IGRS than when OT1 was used as a reference. The OT1 reference signal had a bias in it, due to the off-target position of the laser beam. Prior to the addition of the IGRS, we noted that depending on the location of the beam on OT1, we would obtain differing results. When we added a bias to the IGRS, the controller "behaved", and removed the periodic disturbance as well as centered the beam on target. This effect caused the analytical investigation discussed in Chapter II, and resulted in the ABF of Figure 6. The figures below illustrate the effect of incorrect bias on the nFXLMS controller.

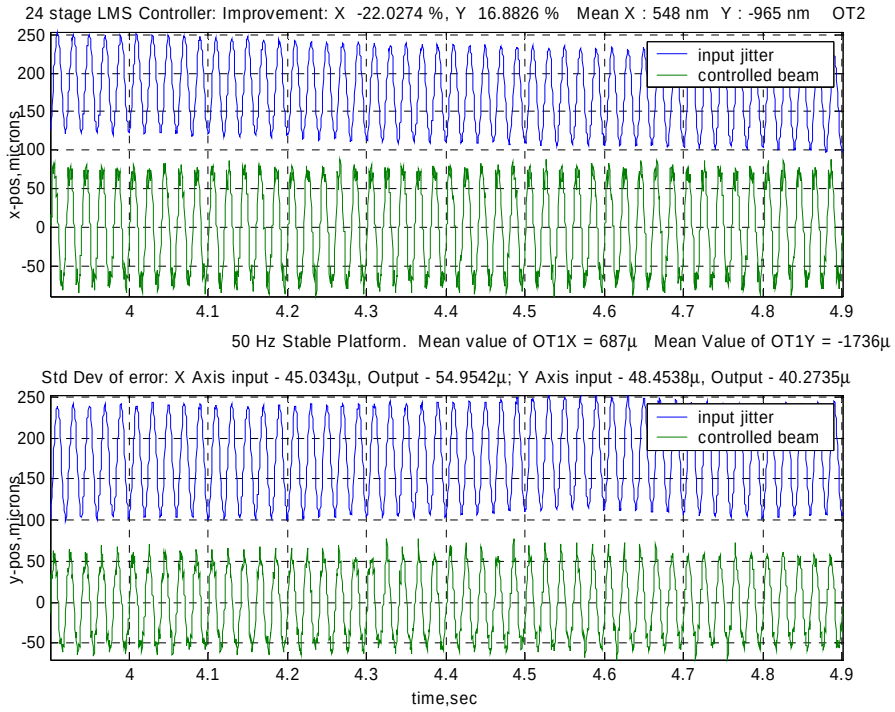


Figure 75 50 Hz disturbance, Stationary Platform, using bias signal from OT1

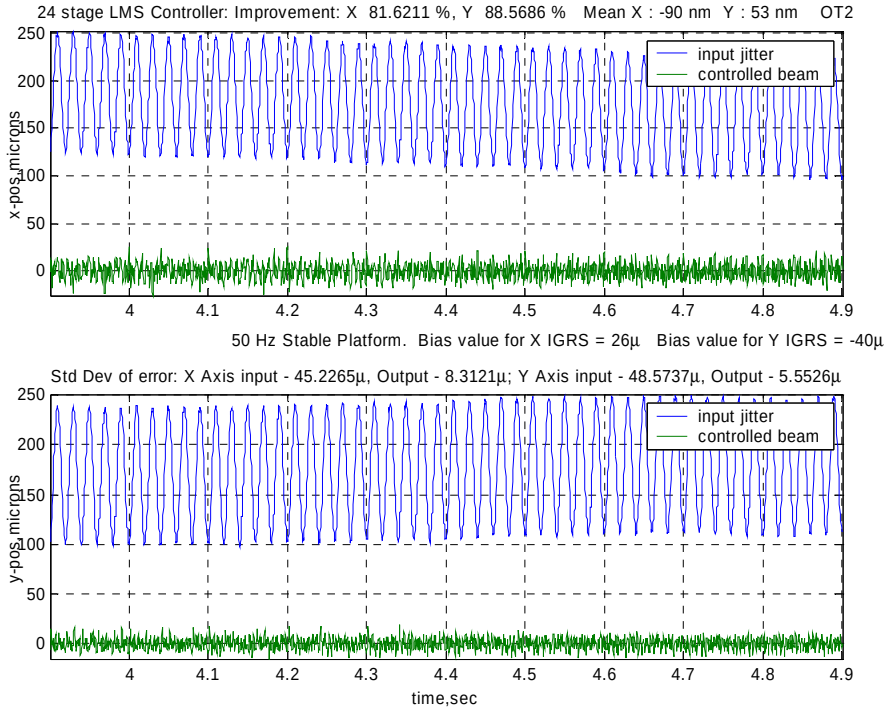


Figure 76 50 Hz disturbance, Stationary Platform, using bias signal from ABF

Unlike the simulation of the 3-mass model, the nFXLMS controller on the LJC testbed would center the beam on the target regardless of whether bias was present or at the correct value in the reference signal. In the simulation of the 3-mass model, the controller would remove the noise, but leave the bias in the output, if no bias was present in the reference signal (see Figure 48). The effect of the bias in the testbed experiments causes a slower convergence to a steady state value, but the bias *is* removed. The reason for this is as yet unknown. It may be that the bias signal on the testbed is varying enough to cause the LMS algorithm to try and remove it, but this has not been verified to date.

a. Proper Bias vs the Use of a Compensator

The nFXLMS controller with proper bias was compared with a regular nFXLMS controller, no bias, using a parallel LQR as a compensator to determine if the addition of bias to the reference signal was better than using a compensator. A 50 and 87 Hz signal was injected by the DFSM and the two controllers used to remove the error in the target signal. The PSD and MSE of the experiments are provided below.

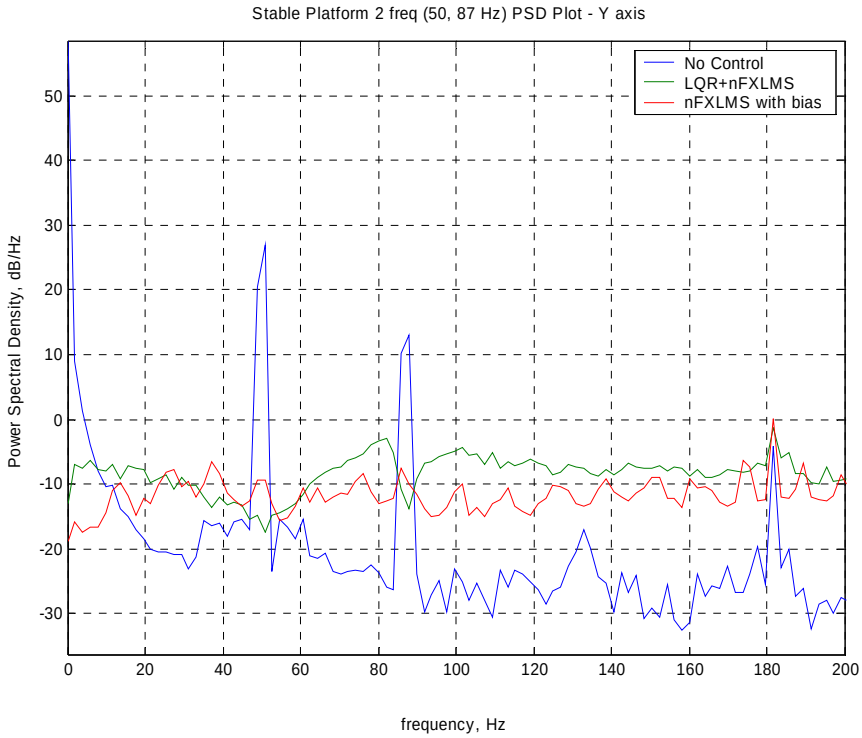


Figure 77 PSD comparison of compensator and biased reference signal

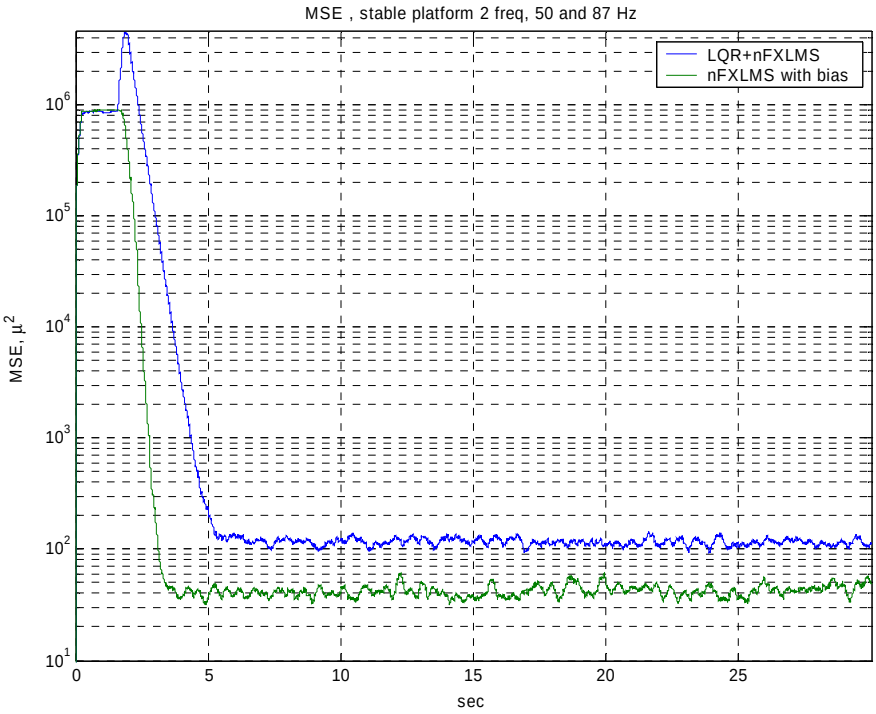


Figure 78 MSE comparison of compensator and biased reference signal

From these figures, we see that the addition of bias to the reference signal results in a similar decrease in the PSD as using a straight compensator, and that the time constant for the biased reference signal is less than that for the parallel adaptive/compensator controller.

3. Random Disturbances

The DFSM was used to inject a random disturbance to the incoming laser beam. This random noise was a 200 Hz, band-limited white noise input to the DFSM, causing a corresponding random disturbance to the beam. This allowed us the ability to observe how the different controllers could handle random disturbances.

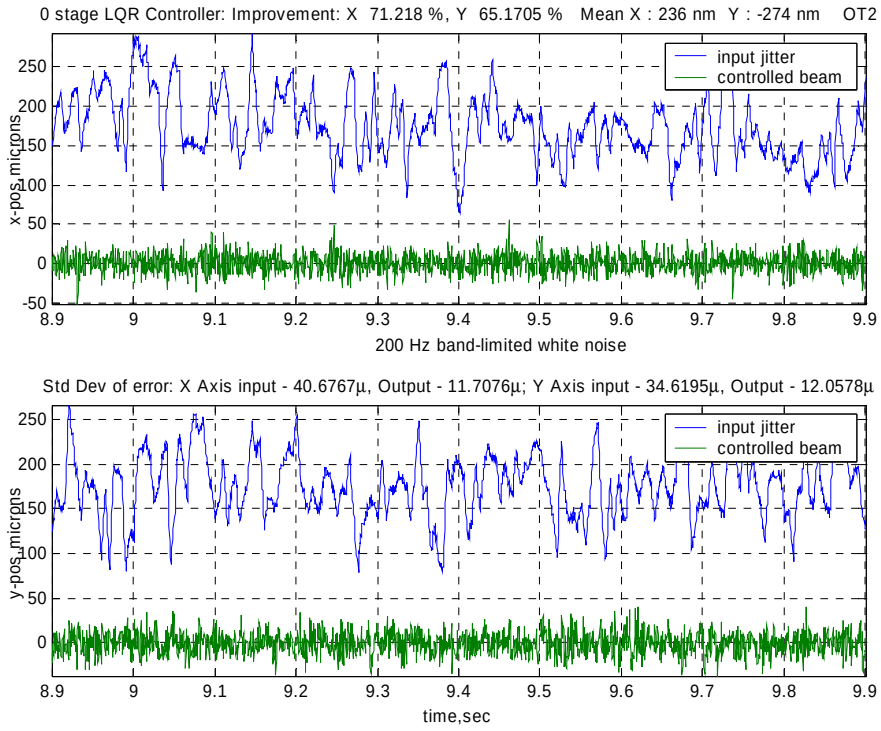


Figure 79 LQR control of 200 Hz band-limited white noise

Figure 79 shows that the LQR controller does a nice job of removing the noise, with a 71% improvement in the standard deviation of the beam pattern in the X position, and a 65% improvement in the Y position. The beam is centered on the detector, within the sensitivity of 500 nm of the sensor. But we must note that, due to the tight beam control desired, the gains are such that the integration to the steady state value requires about 6 seconds (see Figure 82).

The figures below illustrate the nFXLMS and GAL controller response to the random disturbance, using the PSD signal from OT1 as the reference signal. The LMS algorithm does not do as well as the LQR, with only a 13% and 43% improvement in the X and Y position respectively. However, unlike the LQR, the nFXLMS and GAL controller reach their steady state value within 1 second (see Figure 82).

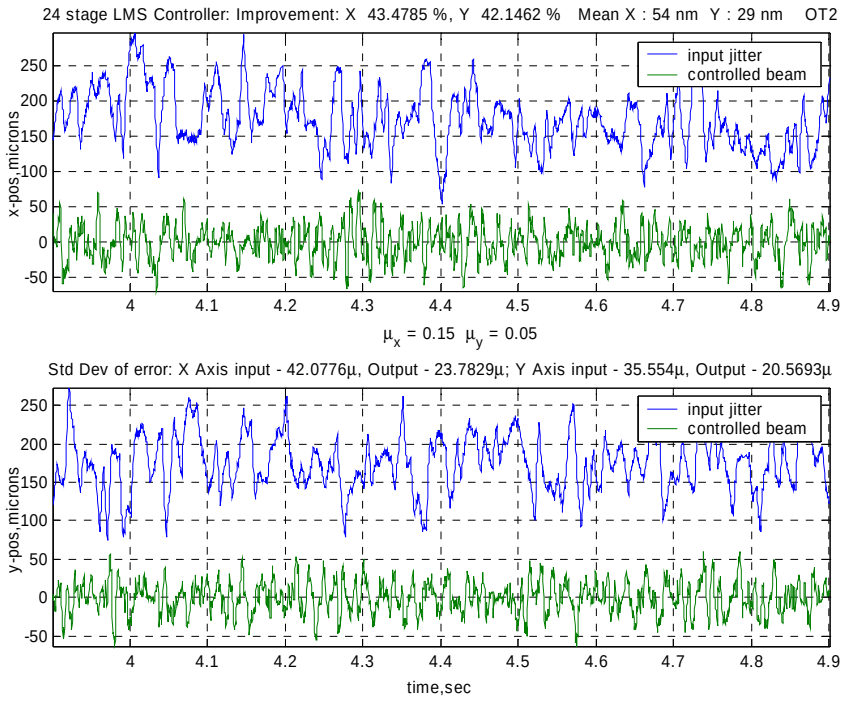


Figure 80 nFXLMS control of 200 Hz band-limited white noise

Likewise, the GAL controller was tested with random noise, and the results are shown below.

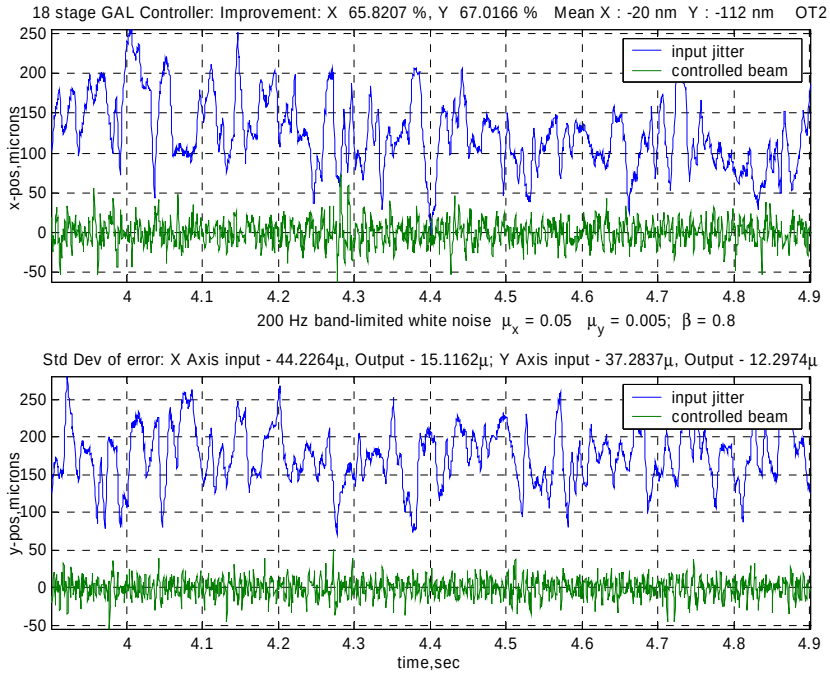


Figure 81 GAL control of 200 Hz band-limited white noise

The GAL controller exhibits the best control of the adaptive controllers for the stationary platform, random noise case, and approaches the LQR controller in effectiveness, as can be seen in the MSE plot below. The value β in the graphs below is the leakage factor discussed in Chapter II.

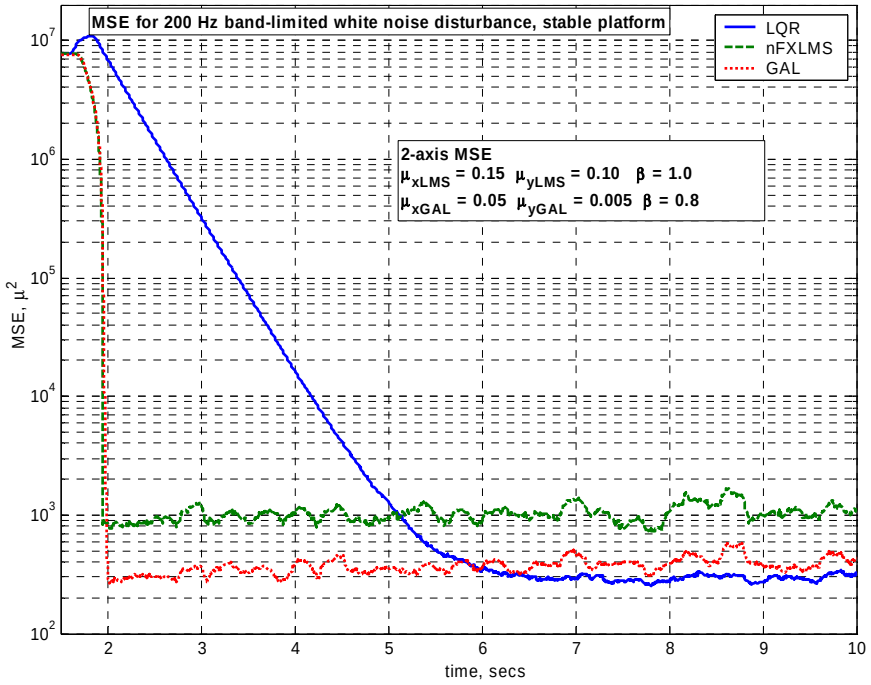


Figure 82 MSE comparison for random noise, stationary platform

4. Multiple Frequencies

The control of multiple frequencies for the stationary platform was attempted. The ADF system was not used for the nFXLMS due to the inability of the system to discriminate and phase shift multiple frequencies. We compared the LQR with the nFXLMS and GAL controllers (both using the OT1 output as the reference signal). The results for two frequencies injected by the DFSM are shown below. The nFXLMS does a better job of removing both frequencies than either LQR or GAL.

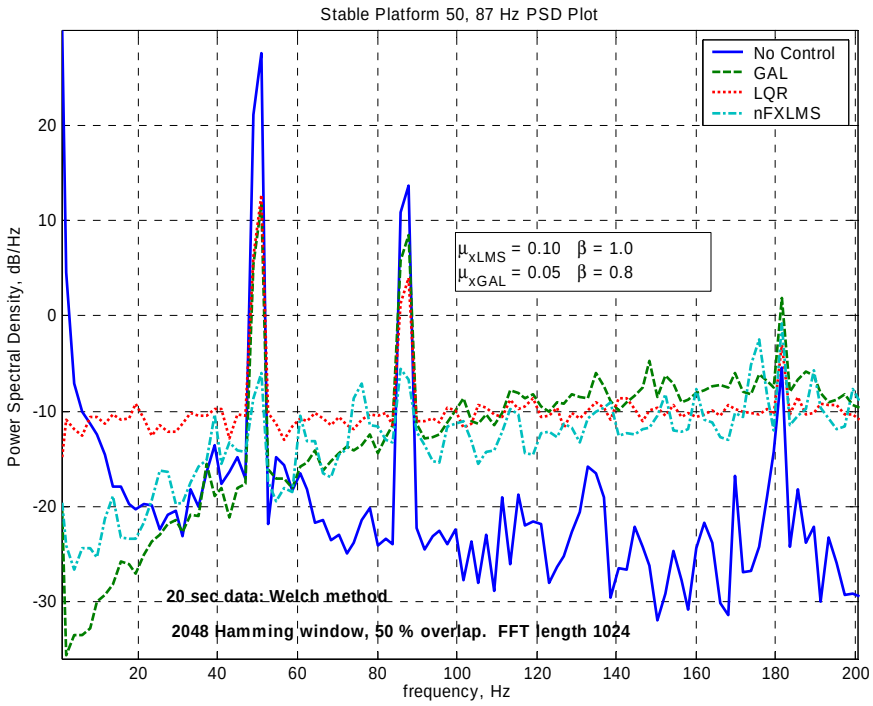


Figure 83 Multiple frequency control – GAL and nFXLMS

C. VIBRATING PLATFORM EXPERIMENTS

Our next series of experiments involved maintaining the DFSM at a zero angle, and shaking the platform with the shaker mounted vertically (Y axis direction for the PSD). Due to the support structure of the platform and the mounting location, the platform moves in both the Y and X directions. The reference signal was synchronized with the disturbance using the either the signal to the shaker or the output from the accelerometer.

1. Periodic Disturbance – Phase Effect

The experiments conducted for the stationary platform configuration were repeated with the platform vibration providing the disturbance. The IGRS was again phase-shifted, and the results are shown below. Bias was added to the reference signal, equivalent to the bias seen at the reference PSD, OT1. The data was calculated from the last 1 second of a 4.4 second run. Similar to the stationary platform case, phase shift has an impact on the convergence properties of the LMS filter for this case.

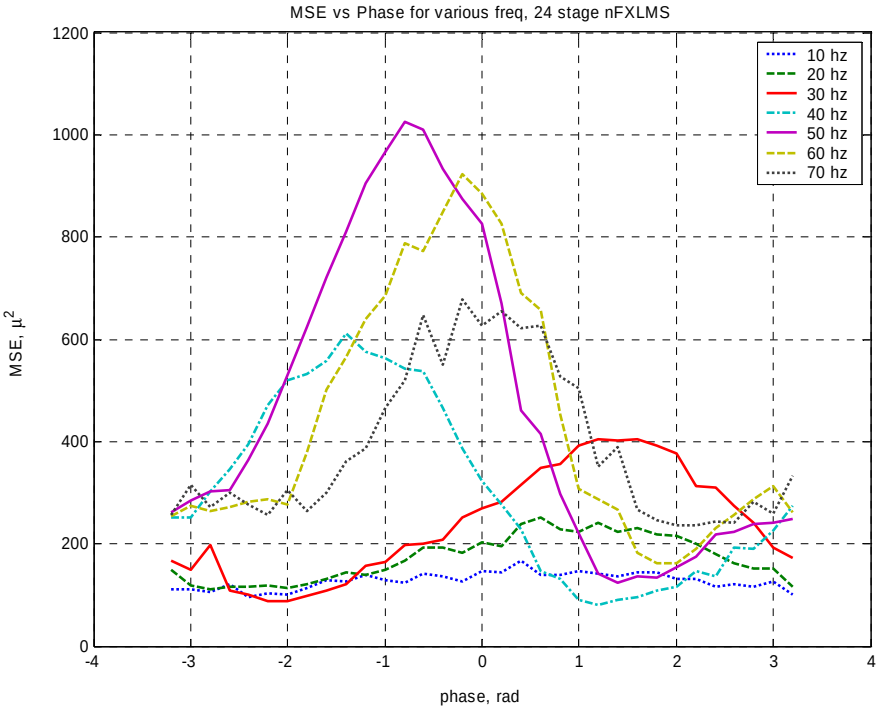


Figure 84 Effect of Phase-shift on MSE for nFXLMS controller, vibrating platform, low frequency

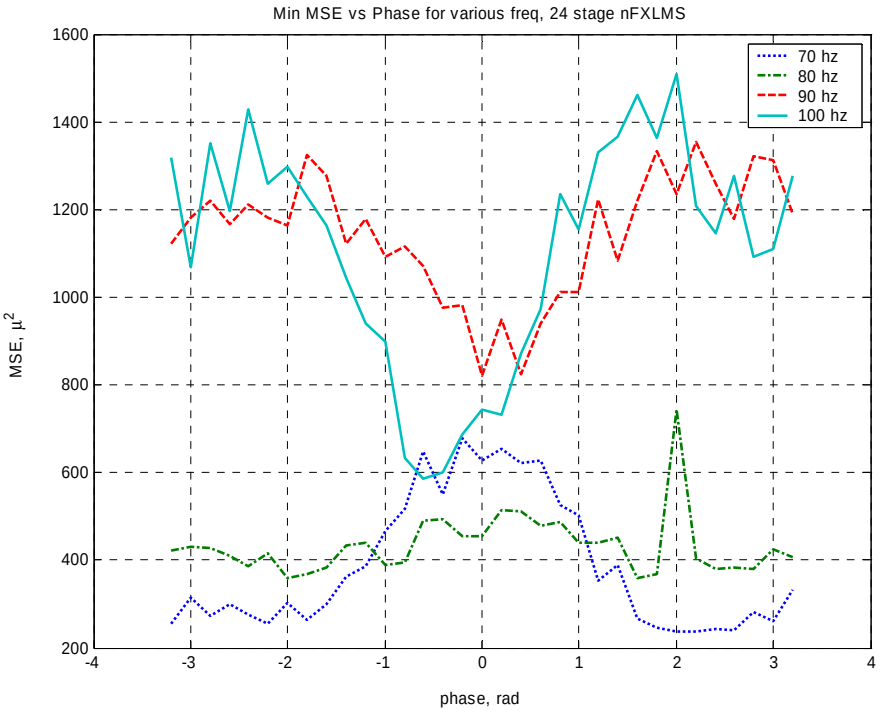


Figure 85 Effect of Phase-shift on MSE for nFXLMS controller, vibrating platform, high frequency

As in the stationary platform case, the GAL controller was tested at various frequencies and compared against the nFXLMS in its various configurations – with OT1 as the reference signal and with the IGRS phase shifted to the optimum value.

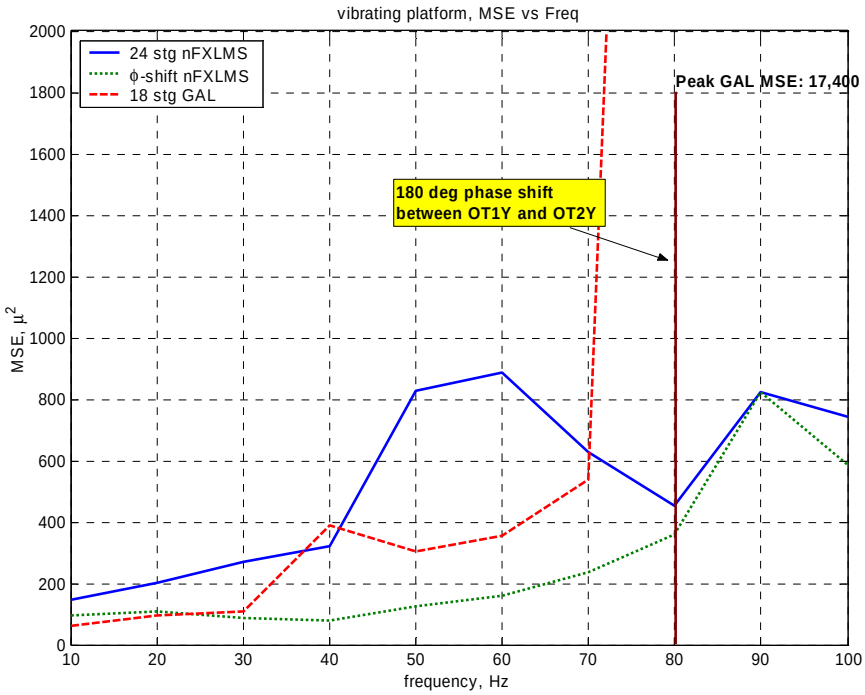


Figure 86 MSE vs. Frequency for nFXLMS, phase-shifted nFXLMS, and GAL controllers, Vibrating Platform

It is easy to see the effect of phase shifts on the GAL controller from Figure 86. From Figure 29, (reprinted below as Figure 87) we see that there is a phase difference between OT1Y and OT2Y starting at about 40 Hz, and a 180 degree difference at 80 Hz. Figure 86 shows that the GAL controller has an increase in MSE at 40 Hz, and actually adds energy to the disturbance at 80 Hz.

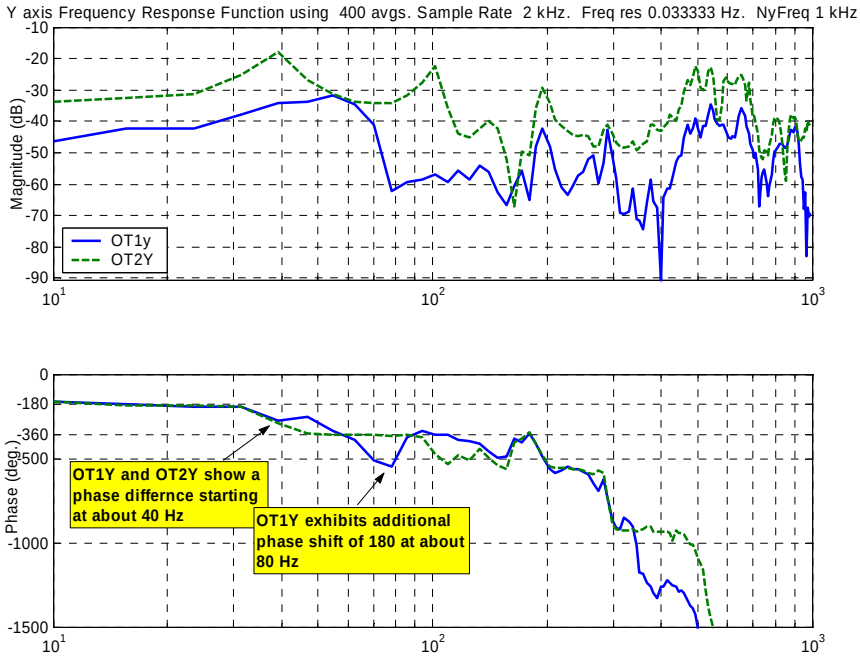


Figure 87 Frequency Response Function, OT1Y and OT2Y

As in the stationary platform case, convergence rates are compared for a single vibration frequency of 50 Hz. Again, the nFXLMS using OT1 as the reference, the nFXLMS with the IGRS, and the nFXLMS with the IGRS and ADF combination are compared against the GAL controller. The figure below shows once again that the GAL controller converges to its steady state value the fastest (approximately 0.4 seconds for the situation). The nFXLMS with IGRS and ADF reaches the steady state value of the GAL controller output in approximately 2 seconds, and is 53% of the steady state value of the GAL controller.

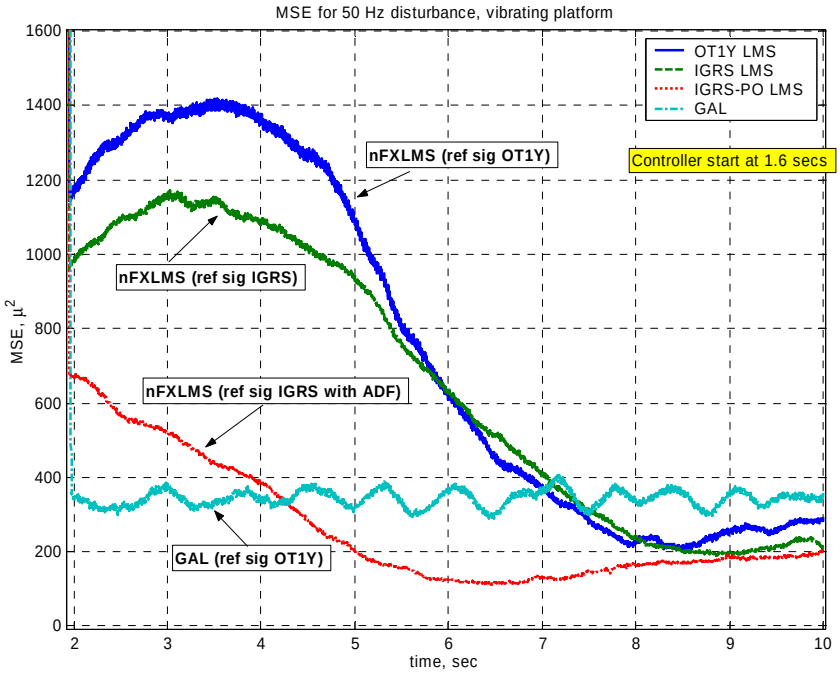


Figure 88 Comparison of controllers for vibrating platform at 50 Hz

2. Periodic Disturbances – Bias Effect

The vibrating platform experiments exhibit the identical response as the stationary platform with respect to the addition of bias to the reference signal. The addition of the proper bias to the reference signal allows for much faster convergence, with the same or better steady state MSE.

3. Random Disturbances

200 Hz band-limited white noise was injected into the shaker. The time domain results for each controller are shown below. These graphs show that for random vibrations, the LQR controller again works best.

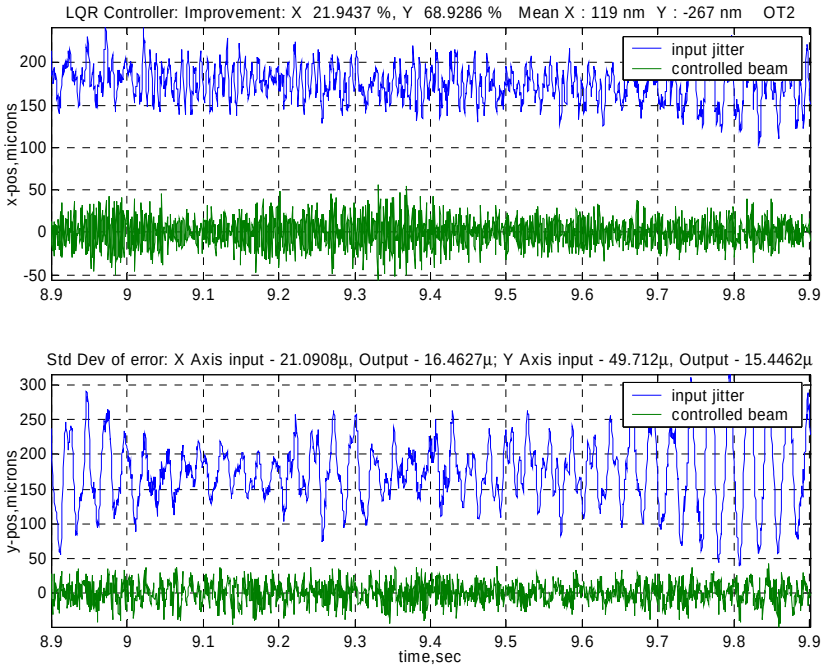


Figure 89 LQR control of random vibrations

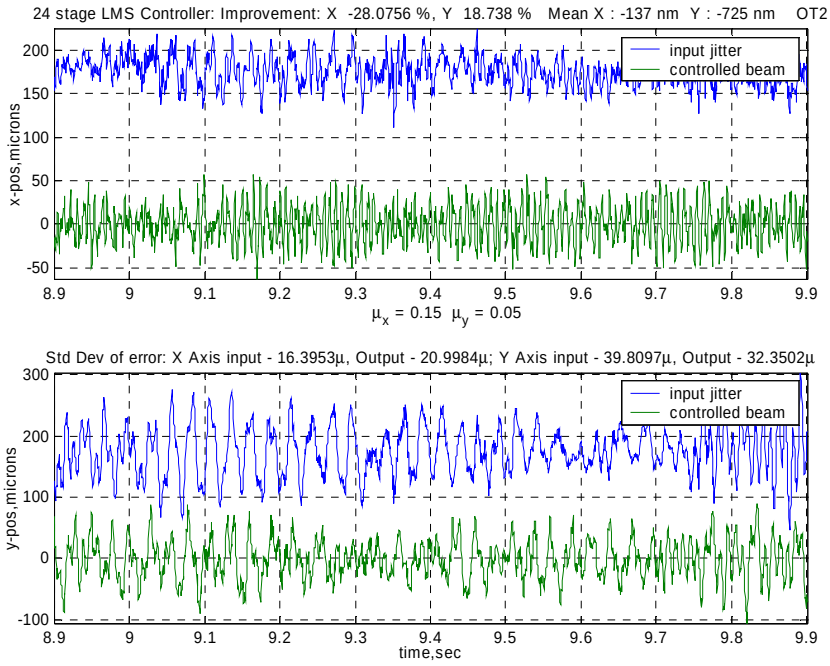


Figure 90 LMS control of random vibrations

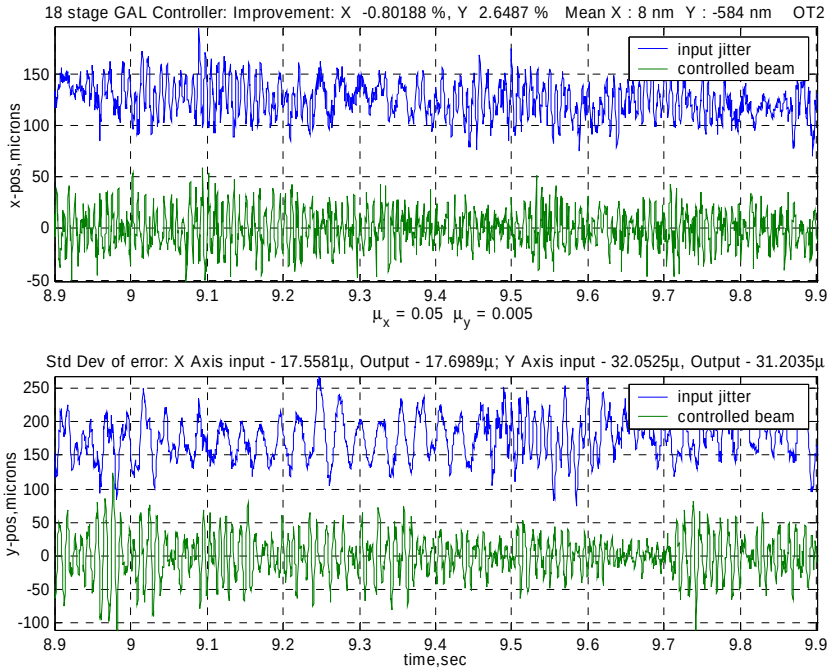


Figure 91 GAL control of random vibrations

4. Multiple Frequencies

The shaker was excited with multiple frequencies to evaluate the ability of the controllers to handle this type of disturbance. Again, 50 and 87 Hz were the input frequencies. As can be seen in the figure below, the GAL controller added energy to the 87 Hz vibration, as well as the 100 Hz vibration from the 2nd folding mirror. Both the nFXLMS and GAL controllers added energy to the resonance shaker frequency of 38 Hz. Based on the results of these experiments and the simulations on the 3-mass model, we determined that an IGRS was necessary for the GAL controller to work in a system in which a phase shift occurs between the reference signal and the error.

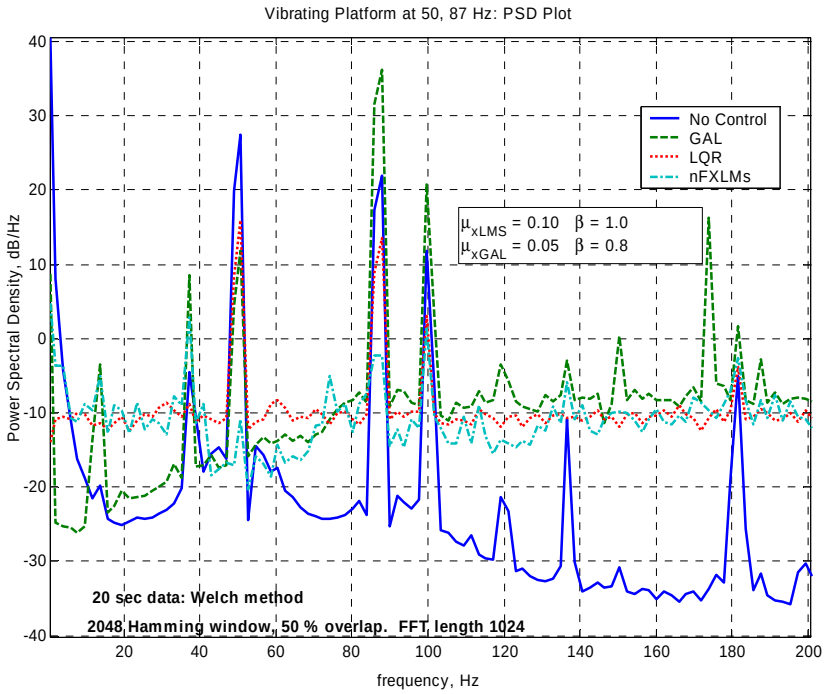


Figure 92 PSD graph of two frequency control, vibrating platform

D. COMBINATION OF VIBRATING PLATFORM AND EXTERNAL DISTURBANCES

The final test of the LJC testbed involves the vibration of the platform at multiple frequencies and the DFSM injecting noise into the laser beam. This experiment is very close to the on-orbit or on-station condition expected for a laser relay platform. The uplink laser beam is corrupted by random disturbances due to thermals in the transmission path, or other causes. The platform itself is disturbed by on-board equipment that causes vibration of the structure, the total disturbance resulting in unwanted jitter. As discussed earlier, this jitter will result in a spreading of the beam, lowering the intensity at the target. Both the LMS and GAL controllers were configured with an IGRS to mitigate the effects of phase shifts between reference and error signals. Additionally, the LMS controller used the ABF filter. The reference signal for the GAL controller was provided a constant bias (optimized for the best response). Due to the complexities inherent in the GAL algorithm, the ABF filter could not be used with the GAL controller without overloading the CPU. The ADF was not used for either algorithm, as it does not contribute significantly to convergence at these adaptation rates.

1. Parallel LQR and nFXLMS Controllers

As shown in section VI.B.3 and VI.C.3 above, the LMS controller does not remove random noise well, but the LQR control is effective with this type of disturbance. By combining the two controllers in parallel, the rapid removal of bias and control of narrowband frequencies by the adaptive algorithms could enhance the removal of noise accomplished by the LQR controller. However, when combining the adaptive controller with the LQR or GAL controller, instabilities result when the LQR controller has an integrator. By setting the integration gain to zero, this instability is removed, and the adaptive controller then acts as a very fast integrator.

2. The Vibration and Noise Experiments

These experiments consist of two narrowband frequencies injected into either the DFSM or the shaker, one on either side of the phase shift between OT1 and OT2 for the vibrating platform case. 50 Hz and 87 Hz were chosen and used for both the DFSM narrowband disturbance and the shaker disturbance for comparison. In addition, the DFSM is used to inject 200 Hz band-limited white noise to simulate a random disturbance in the transmission path for the noise case. Each adaptive controller was used separately and in parallel with the LQR controller. The experiment was performed in four cases, covering the stationary and vibrating platform multiple frequency and noise situations. In each case, the Power Spectral Density plot and Mean Square Error plot are compared for each of the control schemes. The Power Spectral Densities are calculated using the last 4 seconds of data of a 10 second run, by Welch's method. The data is windowed using a 2048 length Hamming Window, and the Fast Fourier Transform (FFT) performed using a 1024 length FFT with a 50% overlap. Results are tabulated for each case. In the following tables, the Input Jitter is the recorded sensor readings at the target in the first second of the run, prior to the controller starting. The standard deviation of each axis for the input is recorded. The Controlled Beam is the one second average of the standard deviation of the beams position for that axis, at the end of the 10 second run. The Mean value of the beam position is the average of the beam's X or Y coordinates over the last one second of the 10 second run, in nano-meters (nm). Note that the minimum sensitivity of the PSD is ±500 nm. The amount that each of the two tones is

reduced is listed (+ indicates energy added to that frequency). The Total MSE is the combined MSE for each axis, using the averaging filter discussed previously. The Time Constant is the time from controller cut-on until the Input MSE is reduced by a factor of 1/e. The PSD plots for each of the cases are for the Y axis only, since the major disturbance axis for the vibration case is in the Y direction. The X axis results are provided in Appendix F.

a. Case 1: Stationary Platform Narrowband Disturbance from DFSM

For this case, the platform remains stationary. 50 and 87 Hz narrowband frequencies are summed and injected into the DFSM to provide the disturbance. The PSD, MSE and tabulated results follow.

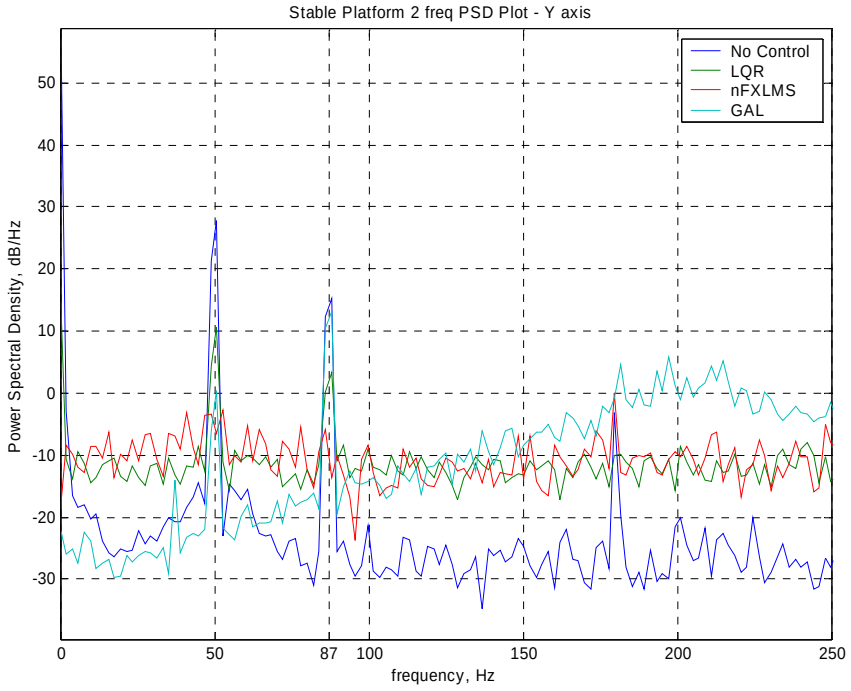


Figure 93 PSD Case 1, DFSM narrowband disturbance

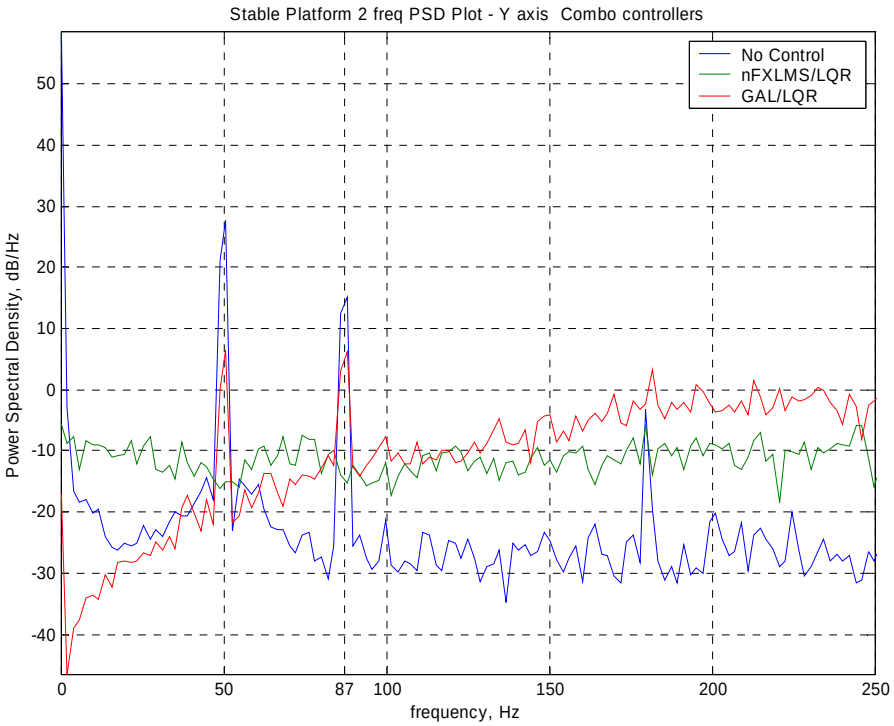


Figure 94 PSD Case 1, Combination Controllers

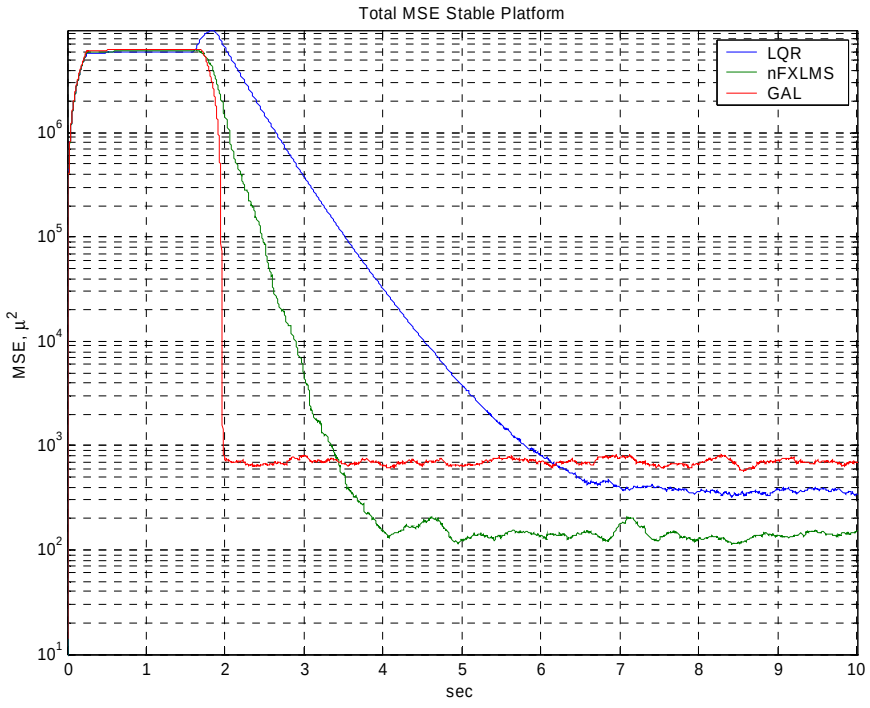


Figure 95 MSE Case 1

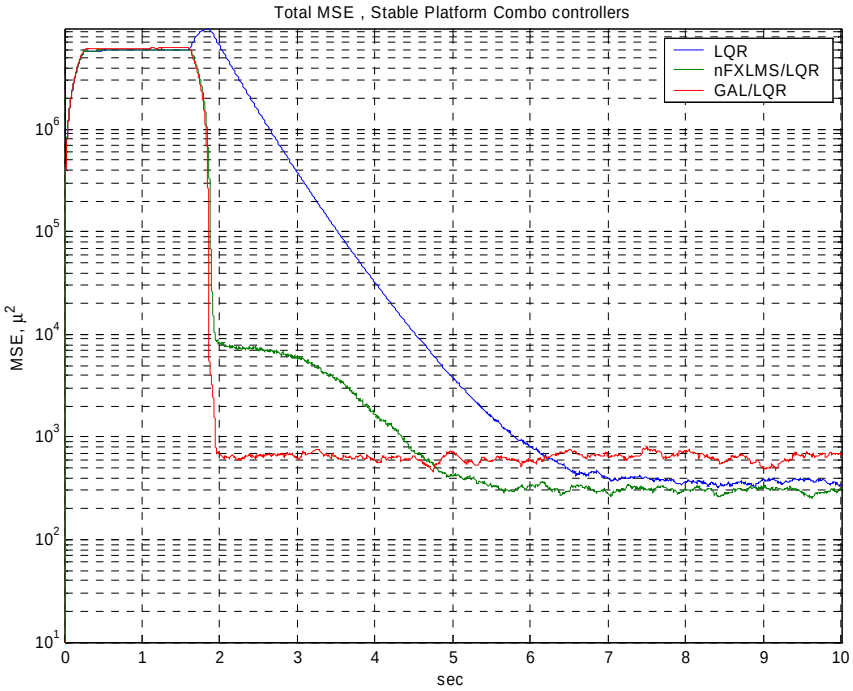


Figure 96 MSE Case 1, Combination Controllers

Controller	LQR		nFXLMS		GAL	
Control Mirror Axis	X-axis	Y-axis	X-axis	Y-axis	X-axis	Y-axis
Input Jitter, Std. Dev, microns	45.4	49.8	45.8	49.9	48.1	52.1
Controlled Beam, Std Dev., microns	10.9	16.0	9.9	6.6	21.1	16.2
No. of stages/order	n/a	n/a	24	24	18	18
Adaptation rate, μ	n/a	n/a	0.10	0.05	0.10	0.06
% reduction in jitter	76.0	68.0	78.3	86.9	56.1	68.9
Mean value of beam position, nm	60	-595	-695	-38	165	-61
db reduction in PSD of 50 Hz	-20.4	-17.0	-30.6	-30.5	-14.4	-27.3
db reduction in PSD of 87 Hz	-14.1	-12.0	-16.6	-20.8	-0.1	-2.0
Total MSE at 10 seconds, micron^2	372.0		142.4		680.1	
Time constant, secs	0.71		0.27		0.26	

Table 3 Case 1 results

Controller	LQR+nFXLMS		LQR+GAL	
Control Mirror Axis	X-axis	Y-axis	X-axis	Y-axis
Input Jitter, Std. Dev, microns	45.3	49.9	47.9	52.0
Controlled Beam, Std Dev., microns	10.1	14.1	19.7	15.5
No. of stages/order	24	24	18	18
Adaptation rate, μ	0.10	0.05	0.10	0.06
% reduction in jitter	77.6	71.8	58.8	70.1
Mean value of beam position, nm	54	52	67	-165
db reduction in PSD of 50 Hz	-38.4	-39.2	-24.5	-21.3
db reduction in PSD of 87 Hz	-24	-23.3	-11.4	-9.1
Total MSE at 10 seconds, micron^2	299.0		596.0	
Time constant, secs	0.16		0.16	

Table 4 Case 1, Combination Controller results

Case 1 shows that the nFXLMS controller works best in removing the stationary platform DFSM narrow band disturbance. The addition of the LQR controller adds to the time required for the controller to approach its steady state value, and results in a higher MSE than for the controller acting alone. This higher MSE for the combination controller is due to the addition of energy by the LQR controller at the higher frequencies, around 600 Hz, as can be seen in the PSD for this case (Figure 97) when expanded to include frequencies to 800 Hz. The LQR controller alone exhibits a similar high frequency response. However, the combination controller does remove the narrowband component better than the nFXLMS controller alone, and centers the beam better on the target. It is noted here, that in each of the 4 cases, the steady state MSE value for the GAL or GAL/LQR controller is rapidly reached, independent of the type of disturbance. For this particular geometry, the GAL controller reaches its steady state value, in each of the cases, in less than 0.4 seconds. As discussed in section II.C, the GAL filter is expected to rapidly converge to its steady state value, and this is borne out by these experiments.

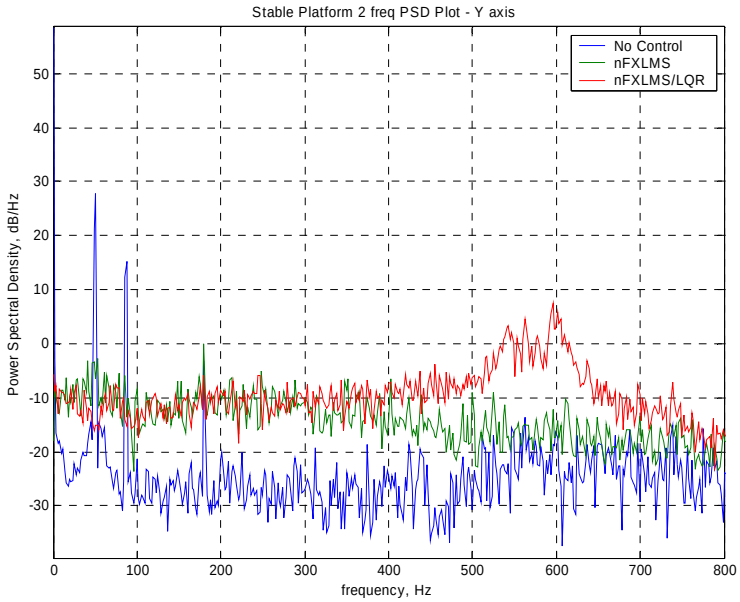


Figure 97 Case 1 Comparison of nFXLMS and nFXLMS/LQR

b. Case 2: Stationary Platform Narrowband Disturbance Plus Noise from the DFSM

In this experiment, the platform is again stationary. 50 and 87 Hz narrowband frequencies are summed and injected with 200 Hz band limited white noise into the DFSM to provide the disturbance. The PSD, MSE and tabulated results follow.

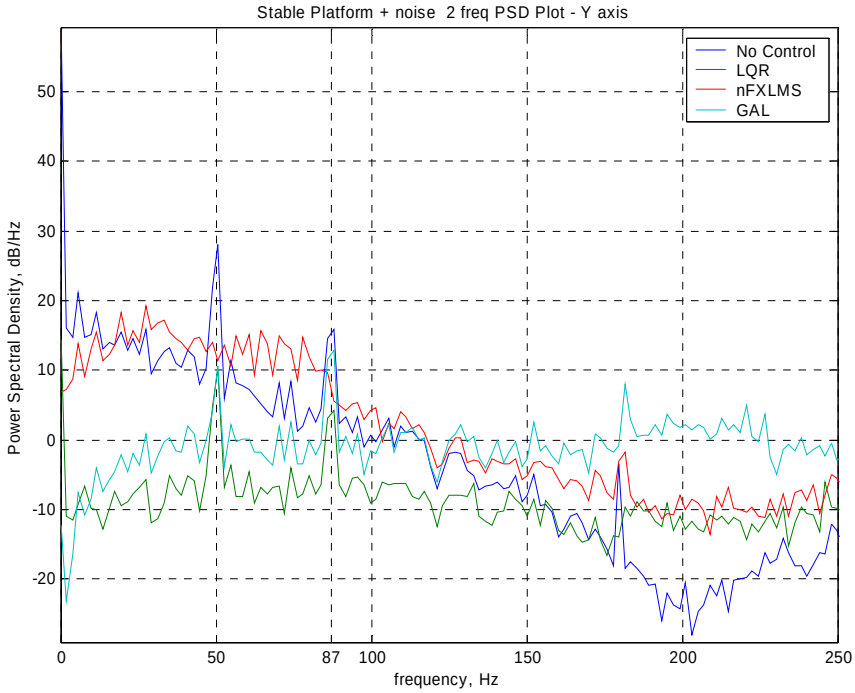


Figure 98 PSD Case 2, DFSM narrowband disturbance plus 200 Hz noise

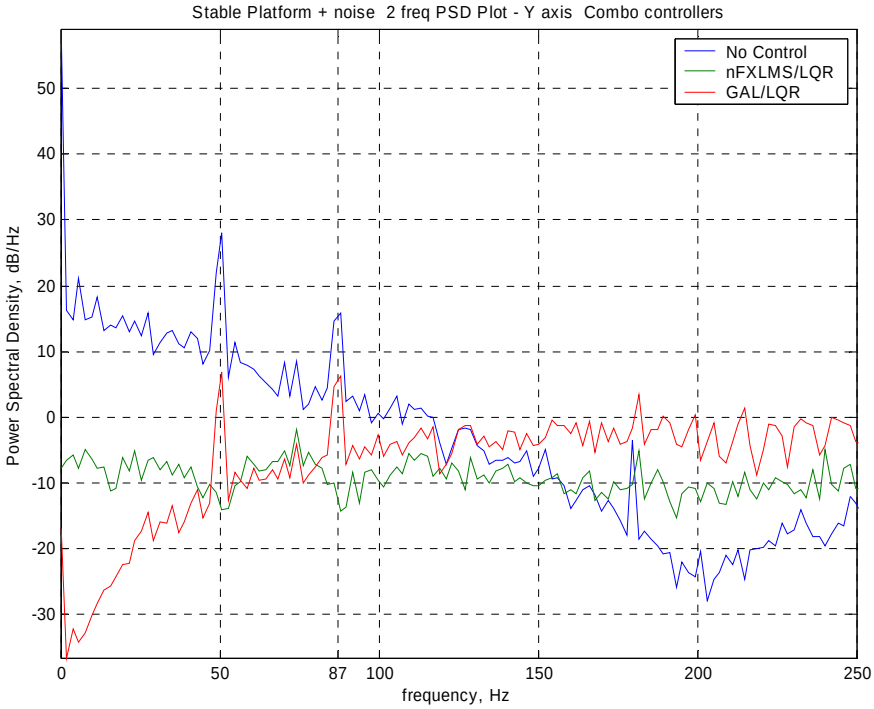


Figure 99 PSD Case 2, Combination Controllers

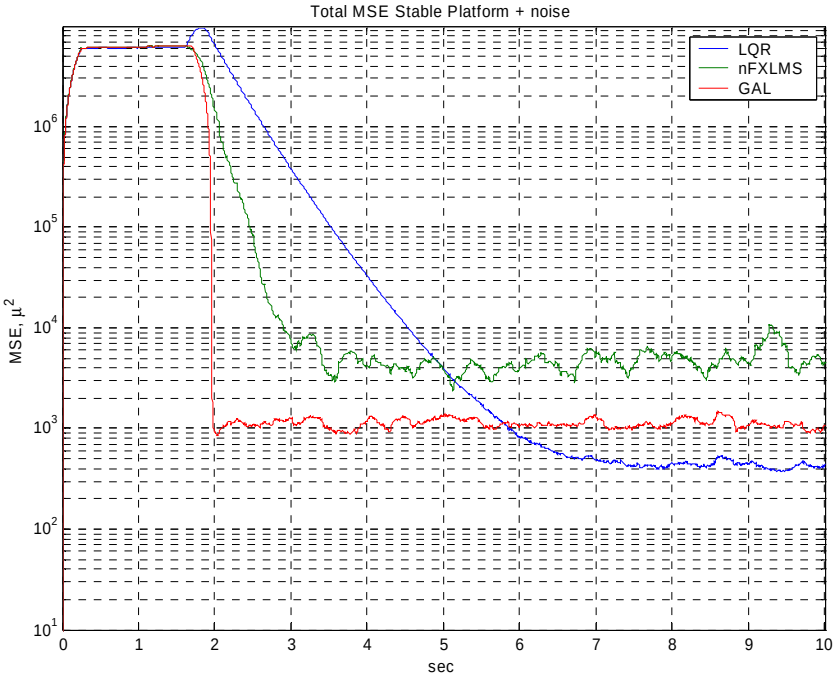


Figure 100 MSE Case 2

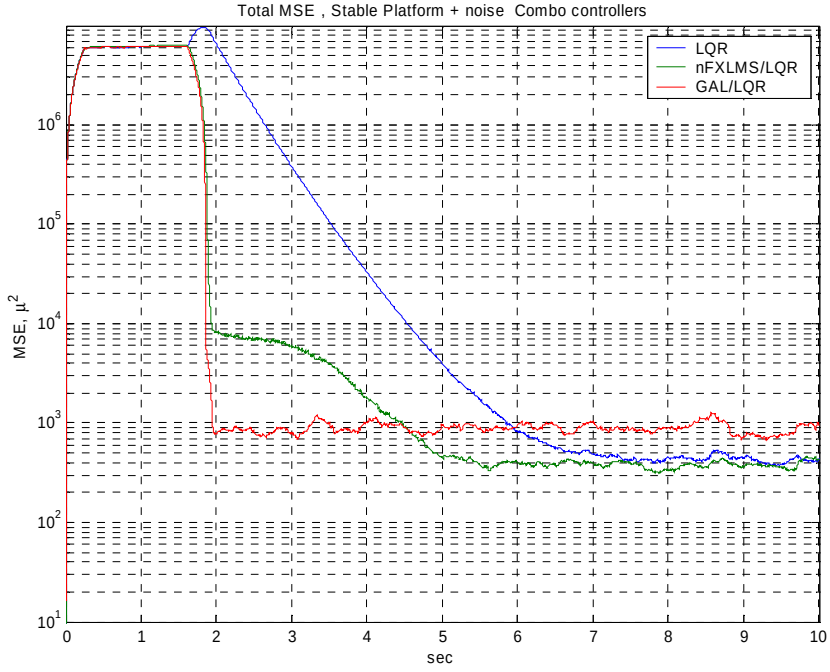


Figure 101 MSE Case 2, Combination Controllers

Controller	LQR		nFXLMS		GAL	
Control Mirror Axis	X-axis	Y-axis	X-axis	Y-axis	X-axis	Y-axis
Input Jitter, Std. Dev, microns	62.3	63.7	63.1	63.2	66.6	65.2
Controlled Beam, Std Dev., microns	12.3	16.3	57.0	52.3	25.3	20.4
No. of stages/order	n/a	n/a	24	24	18	18
Adaptation rate, μ	n/a	n/a	0.1	0.05	0.1	0.06
% reduction in jitter	80.3	74.5	9.7	17.2	62	68.7
Mean value of beam position, nm	87	-515	-1915	3321	-13	-319
db reduction in PSD of 50 Hz	-20.7	-18	-9.9	-12.9	-17.8	-17.7
db reduction in PSD of 87 Hz	-14.3	-12.3	-0.8	-1.1	-0.7	-3.7
Total MSE at 10 seconds, micron^2	416.0		5979.0		1165.0	
Time constant, secs	0.74		0.3		0.25	

Table 5 Case 2 Results

Controller	LQR+nFXLMS		LQR+GAL	
Control Mirror Axis	X-axis	Y-axis	X-axis	Y-axis
Input Jitter, Std. Dev, microns	63.2	63.4	65.7	67.6
Controlled Beam, Std Dev., microns	12.3	15.3	22.8	17.5
No. of stages/order	24	24	18	18
Adaptation rate, μ	0.1	0.05	0.1	0.06
% reduction in jitter	80.6	75.8	65.3	74.2
Mean value of beam position, nm	-63	-23	20	-167
db reduction in PSD of 50 Hz	-34.8	-37.8	-25	-21.5
db reduction in PSD of 87 Hz	-20	-21.1	-12.6	-10.5
Total MSE at 10 seconds, micron^2	386.0		917.0	
Time constant, secs	0.16		0.16	

Table 6 Case 2 Results, Combination Controllers

In this case, we see the detrimental effect of random noise on the nFXLMS controller. The random noise is almost entirely passed through the controller. This can be expected since the reference signal has none of the random component. Even if the reference signal has the random component, as discussed in section VI.B.3 above, the nFXLMS controller does not respond well. Surprisingly, the GAL controller outperforms the nFXLMS controller, again with only the narrowband components in its reference signal. Although there is an increase in energy for this controller above 150

Hz, the random component is almost entirely rejected in the region below 100 Hz, a result not expected based on the analysis of the GAL controller in section II.C. The combination nFXLMS/LQR controller works best for this case. The narrowband components are removed by the LMS algorithm and the random component is removed by the LQR controller. Although the LQR adds energy above 500 Hz as discussed above, the removal of the random component in the range below 150 Hz more than makes up for this in the MSE. The parallel nFXLMS/LQR controller responds much faster than the LQR alone, as can be seen by a time constant of 0.16 versus 0.74 seconds.

c. Case 3: Narrowband Disturbance from the Vibrating Platform

In the Case 3 experiments, we use the inertial actuator to vibrate the platform at the same two narrowband frequencies used in Case 1 and 2 above and compare the results. In this case, the amplitude of the narrowband frequency in the X axis is less, due to the mounting configuration of the inertial actuator in the Y direction. The adaptation rate for the X-axis nFXLMS controller (X-axis of the mirror) was lowered to 0.05 from 0.1 in order to improve performance for the vibrating platform case.

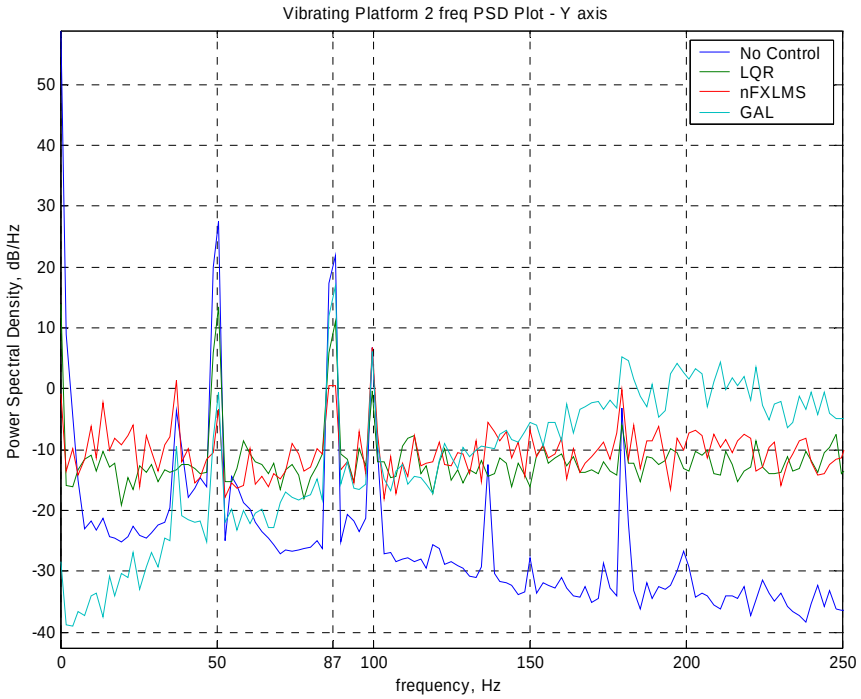


Figure 102 PSD for Case 3: Vibrating Platform Disturbance

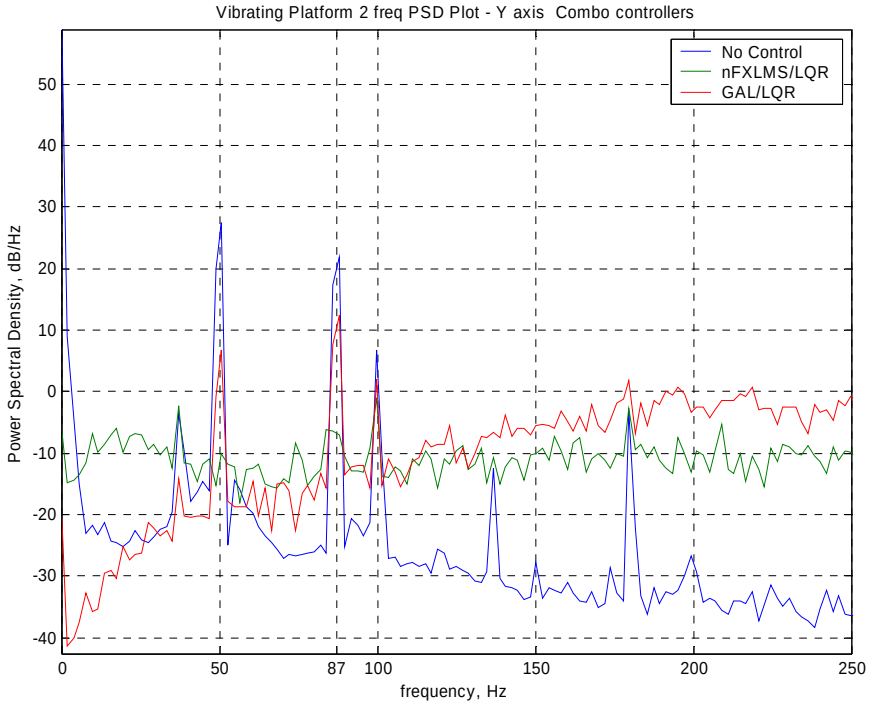


Figure 103 PSD for Case 3, Combination Controllers

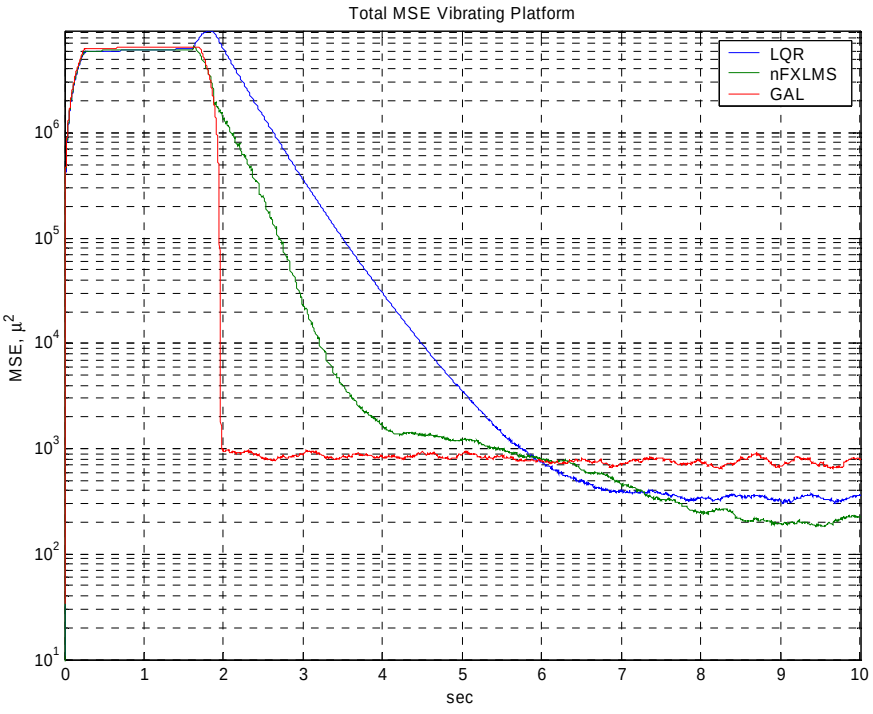


Figure 104 MSE for Case 3

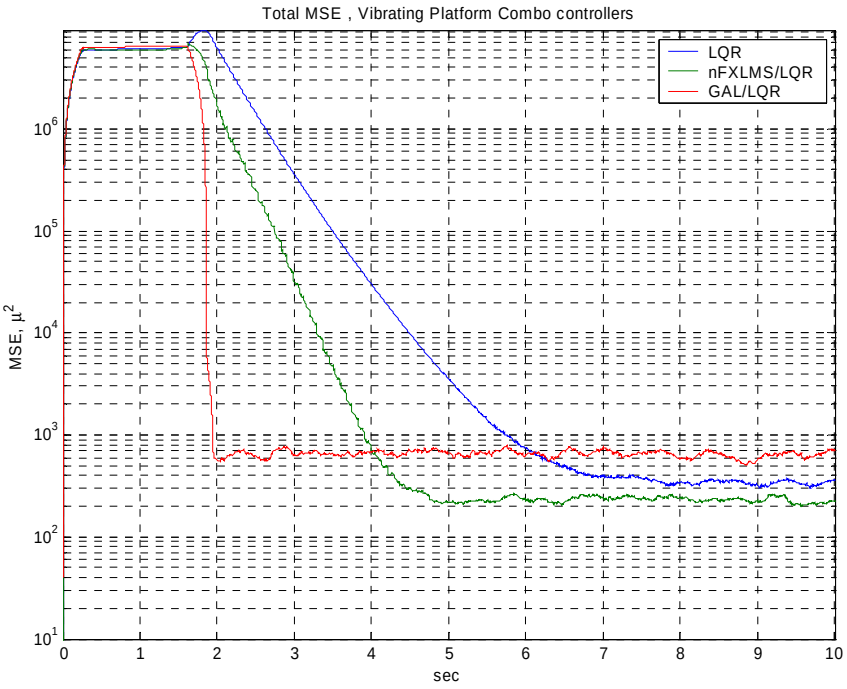


Figure 105 MSE for Case 3, Combination Controllers

Controller	LQR		nFXLMS		GAL	
Control Mirror Axis	X-axis	Y-axis	X-axis	Y-axis	X-axis	Y-axis
Input Jitter, Std. Dev, microns	26.4	53.4	24.5	52.2	24.3	53.0
Controlled Beam, Std Dev., microns	9.9	15.5	12.0	7.8	19.4	19.3
No. of stages/order	n/a	n/a	24	24	18	18
Adaptation rate, μ	n/a	n/a	0.05	0.05	0.1	0.06
% reduction in jitter	62.4	71.1	51.0	85.1	20.5	63.6
Mean value of beam position, nm	18	364	-70	318	9	-26
db reduction in PSD of 50 Hz	-14.6	-14.3	-24.1	-31.2	-15.9	-28.4
db reduction in PSD of 87 Hz	-14.8	-11.0	-2.2	-21.6	-4.2	-5.3
Total MSE at 10 seconds, micron^2	337.8		203.7		848.7	
Time constant, secs	0.72		0.26		0.25	

Table 7 Case 3 results

Controller	LQR+nFXLMS		LQR+GAL	
Control Mirror Axis	X-axis	Y-axis	X-axis	Y-axis
Input Jitter, Std. Dev, microns	27.3	57.6	24.1	53.2
Controlled Beam, Std Dev., microns	9.8	11.4	19.2	17
No. of stages/order	24	24	18	18
Adaptation rate, μ	0.05	0.05	0.1	0.06
% reduction in jitter	64.1	80.1	20.6	68.1
Mean value of beam position, nm	20	318	1	-125
db reduction in PSD of 50 Hz	-29.5	-37.7	-19.5	-21
db reduction in PSD of 87 Hz	-16.0	-28.0	-13.3	-9.8
Total MSE at 10 seconds, micron^2	225.0		681.1	
Time constant, secs	0.34		0.16	

Table 8 Case 3 Results, Combination Controllers

The addition of the IGRS to the GAL controller has mitigated the effect of the phase shift between reference signal and error as can be seen from the decrease in the energy at 87 Hz. As discussed in section VI.C.4, the GAL controller added energy for frequencies above the phase shift between OT1 and OT2. Here, the IGRS effectively shifts the phase 180 degrees for the 87 Hz reference signal, allowing the GAL filter to adapt to lower the energy in this frequency. For Case 3, the nFXLMS controller performs nearly as well as the combination nFXLMS and LQR controller. The combination controller is better in removing the narrowband components than the nFXLMS controller alone as in Case 3.

d. Case 4: Narrowband Disturbance from the Vibrating Platform Plus 200 Hz Noise from the DFSM

In the final case, 200 Hz band limited white noise from the DFSM is added to the disturbance from the inertial actuator. In order to graphically illustrate the case and its results, some time domain plots are provided for the best controllers. First, to demonstrate the effect of the disturbances on the laser beam, a two dimensional view of the experiment beam pattern is shown below. The graph on the right shows the effect of the 50 and 87 Hz vibration of the platform on the beam. The graph on the left shows the effect of the 50 and 87 Hz vibration, plus the random disturbance from the DFSM. Each

asterisk represents the beam's position on the detector at 2 msec intervals. The blue circle in the diagram is drawn at the geometric center of the pattern of the laser beam, and the radius of the circle is one standard deviation of the pattern over 0.1 seconds. Each graph is a snapshot of the beam's position over a 0.1 second interval.

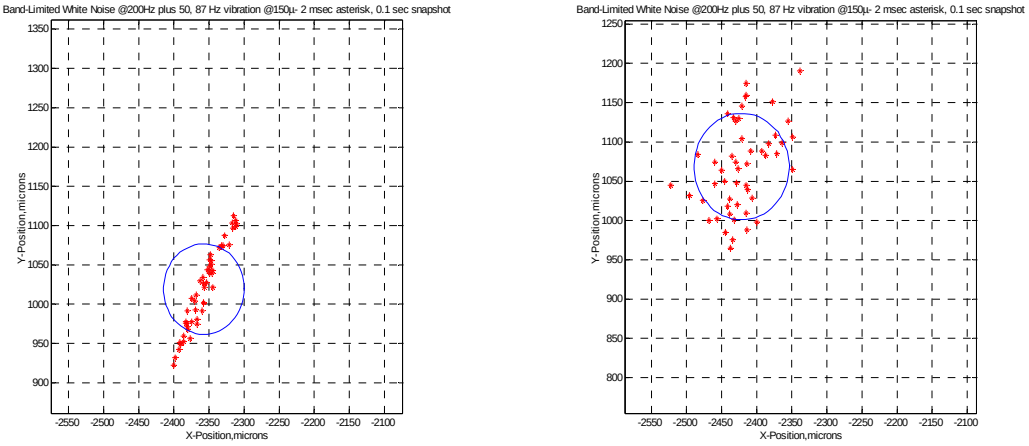


Figure 106 Two Dimensional Beam Pattern

The PSD and MSE results are shown and tabulated below.

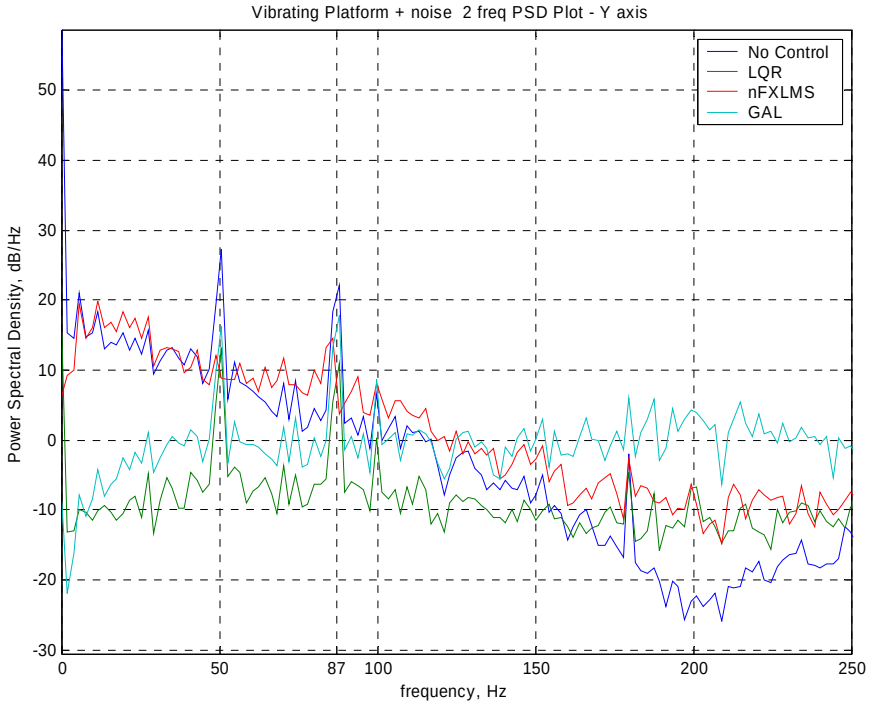


Figure 107 PSD for Case 4, Vibrating Platform plus 200 Hz noise from the DFSM

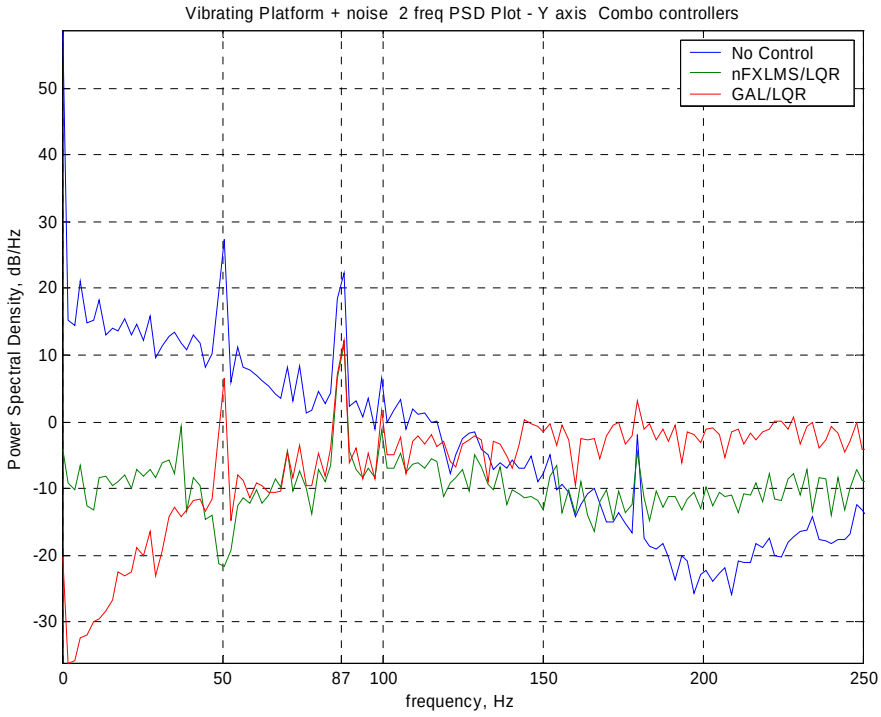


Figure 108 PSD for Case 4, Combination Controllers

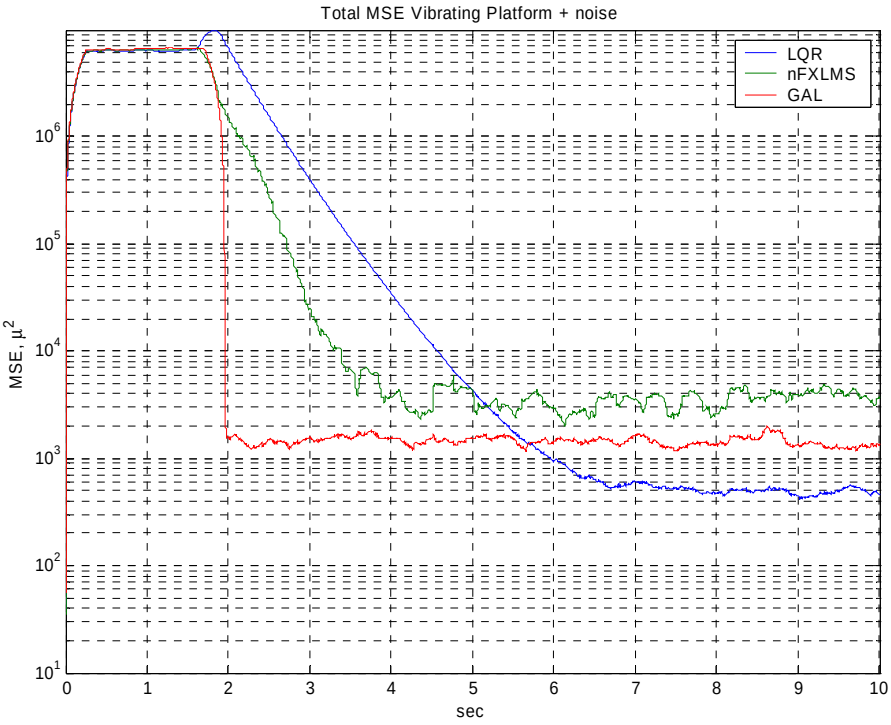


Figure 109 MSE for Case 4

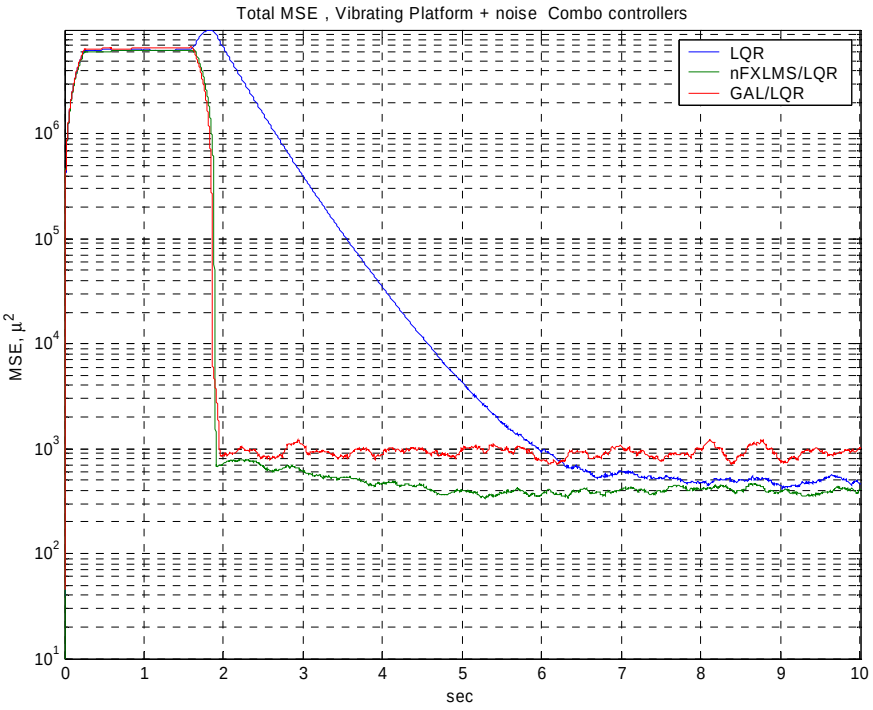


Figure 110 MSE for Case 4, Combination Controllers

Controller	LQR		nFXLMS		GAL	
Control Mirror Axis	X-axis	Y-axis	X-axis	Y-axis	X-axis	Y-axis
Input Jitter, Std. Dev, microns	47.7	68.2	47.2	63.9	51.0	66.7
Controlled Beam, Std Dev., microns	12.4	18.2	43.3	46.4	25.5	25.8
No. of stages/order	n/a	n/a	24	24	18	18
Adaptation rate, μ	n/a	n/a	0.05	0.05	0.1	0.06
% reduction in jitter	73.9	73.3	8.3	27.3	50.0	61.4
Mean value of beam position, nm	165	-763	342	-3063	-84	-373
db reduction in PSD of 50 Hz	-14.1	-15.1	-5.0	-16.1	-7.5	-11.9
db reduction in PSD of 87 Hz	-13.3	-11.2	+1.6	-7.5	-0.5	-4.3
Total MSE at 10 seconds, micron^2	484.8		4054.9		1444.7	
Time constant, secs	0.73		0.26		0.25	

Table 9 Case 4 Results

Controller	LQR+nFXLMS		LQR+GAL	
Control Mirror Axis	X-axis	Y-axis	X-axis	Y-axis
Input Jitter, Std. Dev, microns	44.9	65.5	49.2	66.4
Controlled Beam, Std Dev., microns	12.5	15.3	23.2	19.2
No. of stages/order	24	24	18	18
Adaptation rate, μ	0.05	0.05	0.1	0.06
% reduction in jitter	72.2	76.7	52.8	71.2
Mean value of beam position, nm	2	57	-24	-136
db reduction in PSD of 50 Hz	-23.7	-38.7	-18.5	-21.6
db reduction in PSD of 87 Hz	-13.1	-10.4	-13.4	-9.8
Total MSE at 10 seconds, micron^2	389.2		936.5	
Time constant, secs	0.18		0.16	

Table 10 Case 4 Results, Combination Controllers

The time domain data for the nFXLMS/LQR combination controller is shown below. These results indicate a 72% to 76% reduction in the standard deviation of the input jitter, and the mean value of the beam pattern is well within the measurement sensitivity of the PSD (± 500 nm). The GAL/LQR controller showed a slightly less improvement, 52% to 71% also with the center of the beam pattern within the measurement sensitivity.

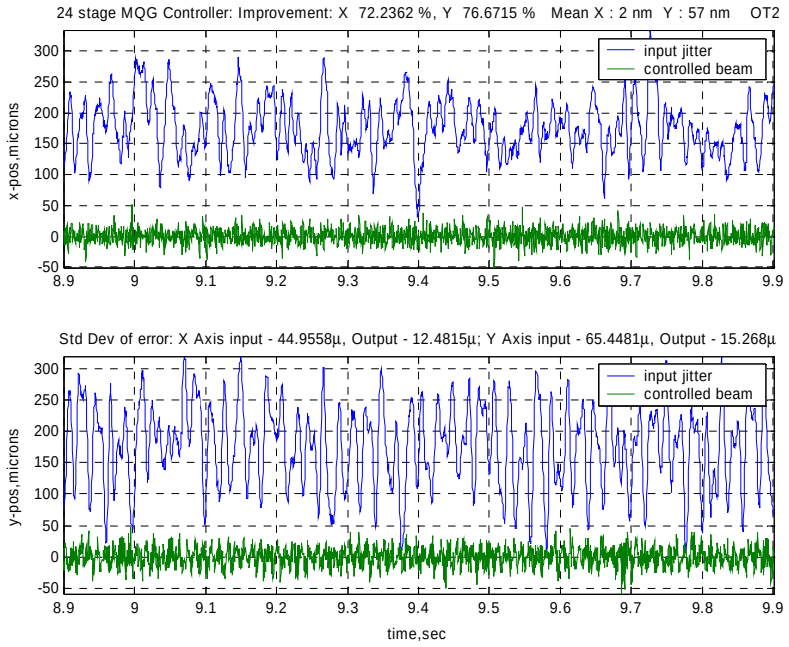


Figure 111 Time Domain results (nFXLMS/LQR) for Case 4

The 2-D views of the nFXLMS/LQR results are provided below. The target is the inner black circle, with a radius of 25 microns. The outer circle has a radius of 50 microns.

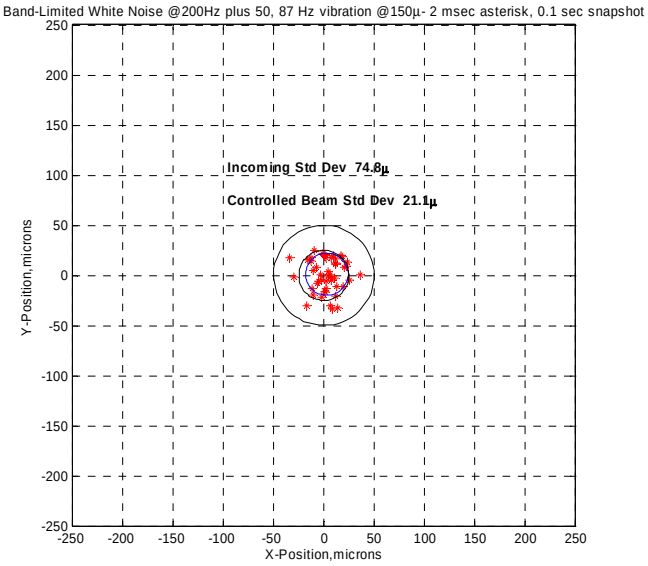


Figure 112 2-D view of the nFXLMS/LQR controller effect on optical beam

The total MSE vs. time graphs are most revealing for this case. In Figure 110 we see the effect of using the adaptive nFXLMS controller in parallel with the LQR controller. By adding the proper bias to the reference signal using the ABF, the nFXLMS/LQR controller is able to rapidly zero the signal, and remove the noise. In combining the two controllers, we get the speed of the adaptive controller in centering the beam and removing the disturbance narrowband frequencies, and the ability of the LQR controller to remove noise, together resulting in a "tight" beam centered on the target.

VII CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

A. CONLUSIONS

As part of this research to develop improved techniques for the control of Optical Beam Jitter, a Laser Jitter Control Testbed was designed and built at the Spacecraft Research and Design Center, Naval Postgraduate School to validate the new techniques experimentally. This Testbed is unique in its ability to experimentally test algorithms for the control of jitter in an optical beam. The testbed has the capability to vibrate the Fast Steering Mirror and associated optics used to correct the disturbances to the beam in a controlled manner, allowing the investigation of different algorithms in suppressing jitter.

This research has shown that the phase difference between the reference signal and the error signal is important in correcting a narrowband disturbance. This phase difference develops since a physical structure is being excited by an inertial actuator or some other vibratory motion. As a result of the excitation, phase differences occur between the reference signal and error signal since these signals are measured at different locations on the structure. By correlating the reference and error signal, improved convergence rates may be obtained, without increasing the adaptation rate. Since the adaptation rate can be lower, we have increased the convergence rate without increasing misadjustment.

The Gradient Adaptive Lattice filter is particularly susceptible to this phase difference. For the GAL filter to control vibrations in a structure, the disturbance must be below the point at which this phase difference occurs, or the reference signal must be phase-shifted to compensate. Thus the selection and modification of the reference signal and an understanding of the physical structure and its frequency response is necessary to properly optimize the GAL filter. The GAL filter does show promise in its ability to correct band-limited white noise that is not structure related – for example, random disturbances that occur as the beam transits the atmosphere.

We have also found that the addition of the correct amount of bias to the reference signal can have the affect of a compensator, allowing the adaptive filter to not only correct the disturbance, but rapidly remove any bias present in the error. This compensation method is faster than applying an integrator to the error signal in an adaptive system in which an LQR compensator is used. The LQR compensator can then be used without an integrator, specifically tuned to remove random noise. This compensator can be placed in parallel with the LMS controller using bias in the reference signal to control a "colored noise" situation, in which there is bias in the error. The resulting combination controller is faster and more effective at removing jitter than either the adaptive system or the LQR controller alone.

In addition to the general conclusions above, several improvements in optical beam jitter control techniques have been made during the course of this research. In particular:

- An Adaptive Delay Filter was developed and implemented to correlate the error and reference signal for the LMS controller. This filter, designed originally for the measurement of time delays in a sampled data system, is used to adaptively delay the reference signal to better correlate the signal with the error.
- The use of an internally generated reference signal, phase shifted to compensate for the difference between the reference measurement point and the error, was developed and implemented to correct the phase difference problem described above for the Gradient Adaptive Lattice filter
- An Adaptive Bias Filter has been developed and implemented to adaptively add bias to the reference signal for adaptive filters. The ABF allows rapid correction of bias errors by an adaptive filter without the use of a compensator.
- A parallel LMS/LQR controller has been developed and implemented to remove bias in the error, correct narrowband disturbances from a vibrating platform supporting the control system, and control noise in an optical beam.

B. RECOMMENDATIONS FOR FUTURE RESEARCH

As a result of the work performed during the course of this research, several areas have arisen that require further study. These include:

- The GAL filter derived for use in the controller for the testbed does not have a provision for the use of a "Filtered-X" version similar to the LMS algorithm. Kuo (1996) has developed a Filtered-X version of a Lattice filter. This version should

be adapted for use in the testbed, and may result in better performance due to the addition of the filtered reference signal.

- The Gradient Based Lattice Filter developed by Chen and Gibson [Chen, 2001], may prove to be superior the GAL filter. This filter was used by Boelitz, et al [Boelitz, 2003] in the ABLE ACE experiment discussed in Section II. Preliminary simulations run on the mathematical model of the LJC testbed using this type filter show that it may outperform both the GAL and the nFXLMS filter in its ability to handle both random and narrowband disturbances. Modifying this filter for use on the testbed and subsequent testing will confirm its usefulness for controlling the laser targeting problem. Although not used on the testbed, the C-code is provided in Appendix E for use in MATLAB/Simulink.
- It is still not fully understood, why the bias present in the error signal is removed when no bias is provided in the reference signal. Simulations and analysis indicate that the bias should remain, and the noise and/or narrowband disturbances should be removed. In the LJC testbed, the presence of bias in the error signal drastically affects the ability of the controller to remove the unwanted disturbance, yet the bias itself is removed. Perhaps it is the variability of this bias, but that must still be proved.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A: FAST STEERING MIRROR DATA

A. NEWPORT FAST STEERING MIRROR

Number of Axes	2 (tip-tilt)
Angular Range from ±10 V	±26.2 mrad (±1.5°), Mechanical[1]
Resolution	≤1 µrad rms, Mechanical[1]
Repeatability	≤3 µrad rms, Mechanical[1]
Accuracy From ±26.2 mrad, 25°C[1,2]	≤0.262 mrad (0.015°), Mechanical[1]
Linearity From ±26.2 mrad, 25°C[1,2]	≤1.0%
Peak Angular Velocity	≥2.5 rad/sec, Mechanical[1]
Peak Angular Acceleration	≥900 rad/sec^2, Mechanical[1]
Closed-Loop Amplitude Bandwidth[2] (-3 dB)	≥550 Hz at 100 µrad amplitude Optional: ≥750 Hz at 100 µrad amplitude
Closed-Loop Phase Bandwidth[2] (60° lag)	≥300 Hz at 100 µrad amplitude
Gain Margin	≥10 dB
Phase Margin	≥45°
Response Flatness[2]	Peaking ≤3 dB
Small Angle Step Response for Steps < 250 µrad, Mechanical[1]	
Rise Time to 90%	≤3.5 msec
Settling Time to 5%	≤5.0 msec
Large Angle Step Response for 26.2 mrad Steps, Mechanical[1]	
Rise Time to 90%	≤3.5 msec
Settling Time to 5%	≤12 msec
Cross-axis Coupling, Static	≤0.1%
Cross-axis Coupling, Dynamic	≤0.1%
Powered Null Offset (Closed-Loop)	±1 mrad, Mechanical[1]
Noise Equivalent Angle (1 Hz to 10 kHz)	≤3 µrad rms
Resolution of Local Position Sensor	≤0.5 µrad
Quiescent Power at FSM Assembly	≤5 W at any angle ±26.2 mrad
Operating Temperature Range[2]	0 to 50°C (32 to 122°F)
Storage Temperature Range	-20 to 55°C (-4 to 131°F)
Warm-up Time for Mirror Stability[2] at 25°C	≤10 minutes
Mirror Thermal Drift[2]	≤5 µrad/°C, Mechanical[1]
Optical Axis Location without base	1.5 in. (38.1 mm) High, Centered Left-to-Right
Mass with base	1.1 lb (0.5 kg)
Envelope without base	3.0 × 3.0 × 2.3 in. [w × h × d] (76.2 × 76.2 × 58.7 mm)
Interconnect Cable Length	9.8 ft (3 m)

Table 11 Newport Fast Steering Mirror Data

Typical FSM-CD100 Controller/Driver Specifications

Command Input and DIT Output	Analog, ±10 V = ±26.2 mrad
Peak Operating Power to Mirror	30 W
Continuous Maximum Operating Power to Mirror	15 W
Thermal Protection	60°C at mirror coil
Current Protection	3 A
Operating Temperature Range	0 to 50°C (32 to 122°F)
Storage Temperature Range	-20 to 55°C (-4 to 131°F)
Power	100-240 V, 50/60 Hz, 2.5 A
Mass	10.7 lb (4.9 kg)
Envelope (without rack mount flanges)	17 × 3.5 × 11.59 in. [w × h × d] (431.8 × 88.9 × 294.4 mm)

Table 12 Newport Fast Steering Mirror Controller/Driver Specifications

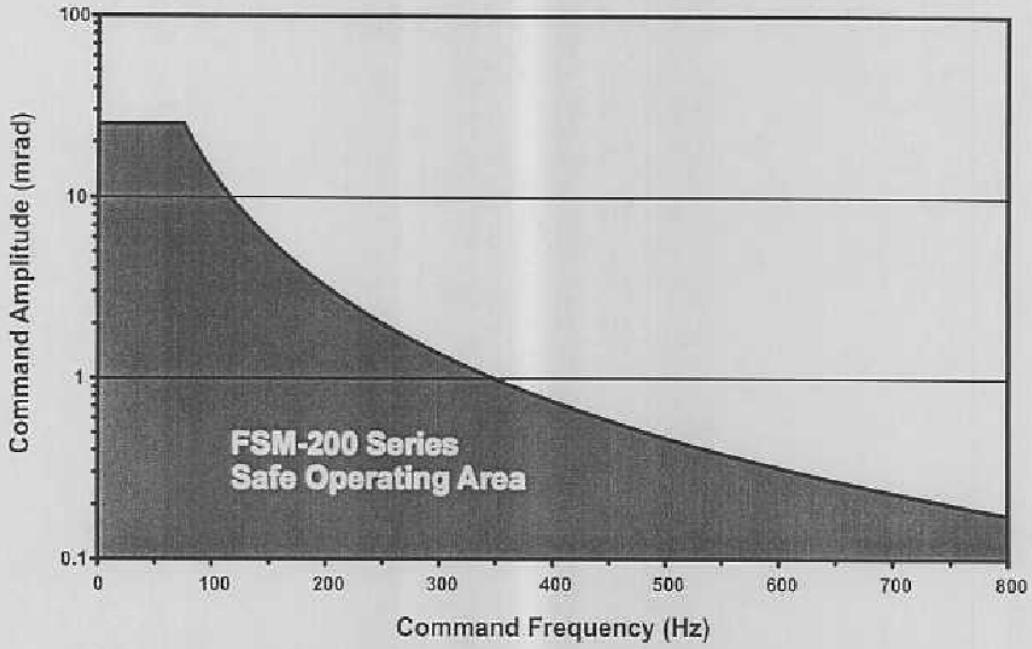


Figure 3: Safe Operating Area for FSM-200 Series fast steering mirrors.

Figure 113 Newport Fast Steering Mirror Safe Operating Area

B. BAKER ADAPTIVE OPTICS FAST STEERING MIRROR

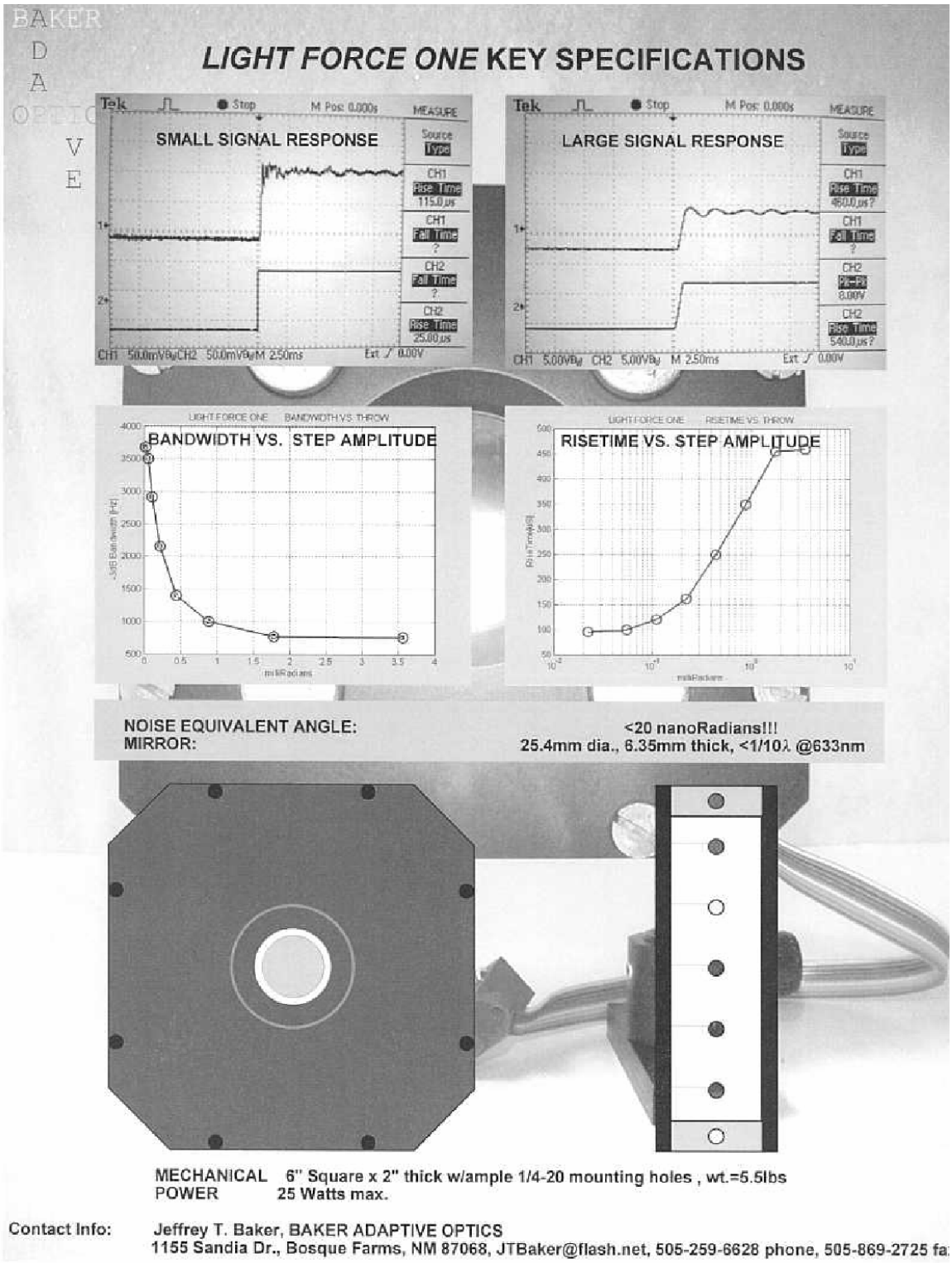


Figure 114 Baker Adaptive Optics – "Light Force One"

C. ONTRAC POSITION SENSING DETECTOR

Model 2L10SP

Model	Active Area (mm)	Responsitivity @940nm	Dark Current nA TYP	Dark Current nA MAX	Noise Current TYP	Noise Current MAX	Capacitance PF@15V TYP	Capacitance PF@15V MAX
2L10SP	10.0x10.0	0.63	100	500	1.3	2.5	90	110

Model	Rise Time 10-90% 15V TYP	Rise Time 10-90% 15V MAX	Reverse Bias V MIN	Reverse Bias V TYP	Reverse Bias V MAX	Detector Resistance (K-ohm) MIN	Detector Resistance (K-ohm) TYP	Detector Resistance (K-ohm) MAX	Thermal Drift ppm/C* TYP	Thermal Drift ppm/C* MAX	Position Non-Linearity +/-% TYP	Position Non-Linearity +/-% MAX
2L10SP	.40	.80	5	15	20	7	10	16	40	200	0.3	0.8

Table 13 OnTrac Position Sensing Detector Specifications

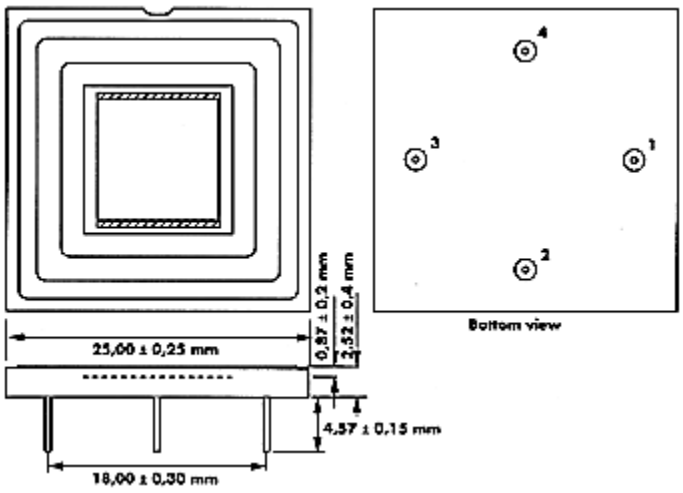


Figure 115 OnTrac Position Sensing Module Diagram

D. CSA INERTIAL ACTUATOR

CSA Engineering, Inc.

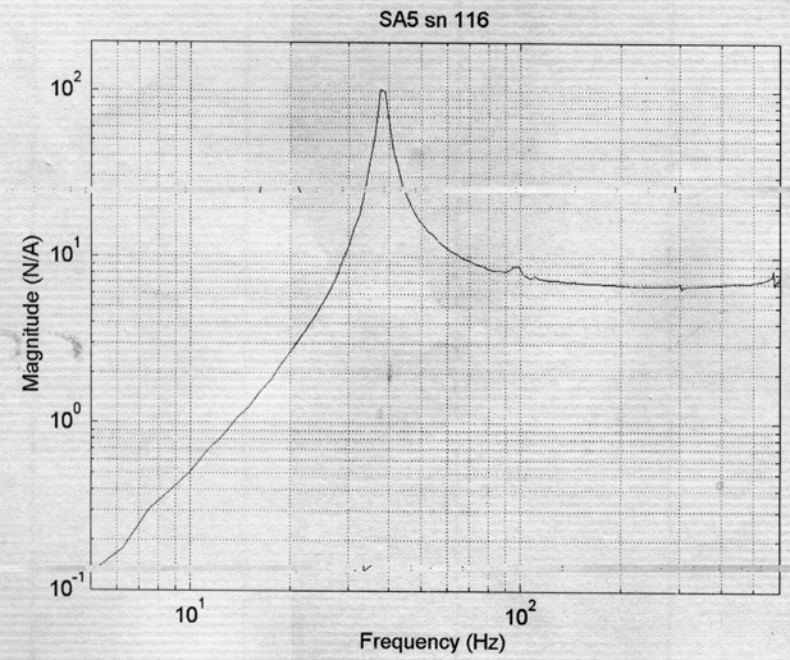
Actuator number: **SA5 - SN 116**

Test date: 2 December 2003
Current drive amplifier

Mechanical interface: single ¼-20, both ends
Gender mate, and mating cable included

Resistance (room temperature with leads): 2.5 Ohms
Inductance (room temperature with leads): 1.03 mH

Maximum continuous drive current, 0-peak: 4.0A



CSA Engineering Inc., 2565 Leghorn St., Mountain View, CA, 94043-1616, USA
www.csaengineering.com 650 210 9000

Figure 116 CSA Inertial Actuator Response Characteristics

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B: SOFTWARE VERSIONS

Software	Version	Function
MATLAB	6.5 R13	Interface Control Computer with xPC Target Computer
SIMULINK	5.0 R13	Model and controller Software interface
RealTime Workshop	5.0 R13	Interface Control Computer with xPC Target Computer
xPC Target	2.0 R13+	Interface Experiment with Control Computer
dSPACE	Release 3.3	Interface Experiment with Disturbance Computer
Microsoft Visual C++	6.0	Compiler
DSP Blockset	5.0 R13.0.1	MATLAB Toolbox
Control System Toolbox	5.2 R13.0.1	MATLAB Toolbox
Windows XP	5.1 Build 2600 Service Pack 1	Control and Disturbance Computer Operating System

Table 14 Software Versions

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C: CALIBRATION PROGRAM

A. CALIBRATION PROGRAM

```
%  file to calibrate mirrors and detectors
%
%  Mirror gains will be placed in a file in the working directory called
%  "mgains.mat"
%  Output figures are in directory "C:\Watkins\MATLAB Files\output figs\
%  This Program uses model calibration_control
%-----------------------------------------------------------
%  Sample Time for model:  do not set less than 0.0005
%-----------------------------------------------------------
Ts=0.0005;
%-----------------------------------------------------------
%   Initiate input/output variables
%-----------------------------------------------------------
ot1x=1;    ot1y=2;   ot2x=5;  ot2y=6;  ot3x=7;   ot3y=8;
nxin=10;     nyin=9;    bxin=4;  byin=3;
signl=-1;  const=1;  ramp=1;
tstng=-1;
step=2;    step_time=0.25;
%-----------------------------------------------------------
%  distances in mm from mirrors to detectors
%-----------------------------------------------------------
d1=1585;   % Dist from DFSM Mirror to OT1
d2=370;   % Dist from RFSM Mirror to OT3
d3=1245;  % Dist from RFSM Mirror to OT2
d11=1370;  % Dist from DFSM Mirror to OT3
%-----------------------------------------------------------
%   Initiate switches
%-----------------------------------------------------------
x_test=ramp;     y_test=ramp;
bx=0;            by=0;
b_test_x=const;  b_test_y=const;
b_x_zero=0;      b_y_zero=0;
n_x_zero=0;      n_y_zero=0;
n_test_x=const;  n_test_y=const;
rampstep=step;   finval=1;
bak_x=1;         bak_y=1;
newx1=1;         newy1=1;
newx2=1;         newy2=1;
ramp_slope=10;   ramp_initial=-10;
%-----------------------------------------------------------
tol=0.001;          %tolerance for determining zero position
%                   do not set less than 0.001
%-----------------------------------------------------------
%   Initiate Program
%-----------------------------------------------------------
disp('energize Laser, OT301 amps, mirrors. Float Table if desired');
disp('allow 20 min stabilization time for accurate readings');
curr_dir=cd;
cd('C:\Watkins\Matlab Files\Calibration programs');
disp('If Calibration model is open, it must be manually compiled before running');
disp('If model is open and not built, exit this program by typing x and enter')
reply1 = input('otherwise press Enter to continue       ','s');
if ~isempty(reply1)
    disp('Manually Build calibration_control model, or exit model, and restart
calibrate');
    error('exiting program');
end
%-----------------------------------------------------------
%   Check connections
%-----------------------------------------------------------
% is connection with target working?
```

```
   if ~strcmp(xpctargetping, 'success')
     error('Connection with target cannot be established');
   end
   % is calibration_control already open?
   systems = find_system('type', 'block_diagram');
   if isempty(strmatch('calibration_control', systems, 'exact'))
     mdlOpen = 0;
     load(xpc,'calibration_control');
   else
     mdlOpen = 1;
   end
   %-------------------------------------------------------
   %       Obtain values for parameters from model
   %-------------------------------------------------------
   tg=xpc;

   b_test_x_par=getparamid(tg,'Switching/b test x','Value');
   b_test_y_par=getparamid(tg,'Switching/b test y','Value');

   rampstep_par=getparamid(tg,'Test Signals/rampstep','Value');
   ramp_slope_par=getparamid(tg,'Test Signals/Ramp/Step','After');
   ramp_initial_par=getparamid(tg,'Test Signals/Ramp/Constant1','Value');
   finval_par=getparamid(tg,'Test Signals/Step1','After');

   n_test_x_par=getparamid(tg,'Switching/n test x','Value');
   n_test_y_par=getparamid(tg,'Switching/n test y','Value');

   b_x_zero_par=getparamid(tg,'Switching/b x zero','Value');
   b_y_zero_par=getparamid(tg,'Switching/b y zero','Value');

   n_x_zero_par=getparamid(tg,'Switching/n x zero','Value');
   n_y_zero_par=getparamid(tg,'Switching/n y zero','Value');

   x_test_par=getparamid(tg,'Test Signals/zero','Value');
   y_test_par=getparamid(tg,'Test Signals/zero1','Value');

   bak_x_par=getparamid(tg,'bak_x','Gain');
   bak_y_par=getparamid(tg,'bak_y','Gain');

   newx_par=getparamid(tg,'new_x','Gain');
   newy_par=getparamid(tg,'new_y','Gain');

   clc
   %-------------------------------------------------------
   %   Set switches in model for zero input values
   %   and start model to obtain and plot data
   %-------------------------------------------------------
   set_switches
   +tg
   pause(1.5);
   -tg
   tt=tg.Time;
   oo=tg.Output;
   set(figure, 'Name', 'zero input', 'NumberTitle', 'off');
   plot(tt,oo(:,ot1x),tt,oo(:,ot1y),tt,oo(:,ot2x),tt,oo(:,ot2y),tt,oo(:,ot3x),tt,oo(:,ot3
y)),grid,zoom
   xlabel('time')
   ylabel('amplitude, V')
   legend('OT1 X','OT1 Y','OT2 X','OT2 Y','OT3 X','OT3 Y')
   title('zero input noise, OT1,OT2 and OT3')
   saveas(gcf,'C:\Watkins\MATLAB Files\output figs\zero_input_noise.fig')
   avgx=mean(oo(1000:2000,ot1x));
   avgy=mean(oo(1000:2000,ot1y));
   disp('Zero value should be + or - 0.1V. Is the zero Satisfactory?');
   reply = input('Enter n to manually adjust, otherwise, hit Enter to continue: ','s');
   %-------------------------------------------------------
   %   Loop for adjustment of zero if necessary
   %-------------------------------------------------------
```

```
   if ~isempty(reply)
       tg.StopTime=9999;
       +tg
       xpcscope
       disp('set mirrors to center positions on OT1, OT2 and OT3,then Enter')
       disp('OT1 is scope 1, OT2 is scope 2, OT3 is scope 3')
       disp('Load Scope "align_scopes" if desired')
       user_entry = input('hit Enter when aligned and ready to proceed');
       -tg
       b_test_x=const;  b_test_y=const;
       rampstep=step;
       set_switches
       +tg
       pause(1.5);
       -tg
       clear tt oo
       tt=tg.Time;
       oo=tg.Output;
       set(figure, 'Name', 'zero input, adjusted', 'NumberTitle', 'off');

plot(tt,oo(:,ot1x),tt,oo(:,ot1y),tt,oo(:,ot2x),tt,oo(:,ot2y),tt,oo(:,ot3x),tt,oo(:,ot3y))
,grid,zoom
       axis([-inf inf -0.1 0.1])
       xlabel('time')
       ylabel('amplitude, V')
       legend('OT1 X','OT1 Y','OT2 X','OT2 Y','OT3 X','OT3 Y')
       title('zero input noise, OT1,OT2 and OT3')
       saveas(gcf,'C:\Watkins\MATLAB Files\output figs\zero_input_noise.fig')
       avgx=mean(oo(1000:2000,ot1x));
       avgy=mean(oo(1000:2000,ot1y));
   end
   %-----------------------------------------------------------
   %   Begin test of mirrors and obtain calibration data
   %-----------------------------------------------------------
   %   Set the values for a ramp test of DFSM
   rampstep=ramp;
   ramp_slope=10;    ramp_initial=-10;
   b_test_x=tstng;    b_test_y=const;
   %-----------------------------------------------------------
   % ramp DFSM in theta x to get OT limits
   %-----------------------------------------------------------
   set_switches
   +tg
   pause(2.5);
   -tg
   clc
   clear tt oo
   tt=tg.Time;
   oo=tg.Output;
   set(figure, 'Name', 'Baker Ramp X, OT1', 'NumberTitle', 'off');
   plot(tt,oo(:,ot1x),tt,oo(:,ot1y)),grid,zoom
   xlabel('time')
   ylabel('amplitude, V')
   legend('OT1 X','OT1 Y')
   title('Baker ramp response,theta X, OT1')
   saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Baker ramp theta X, OT1.fig')
   %-----------------------------------------------------------
   %   Calculate min and max values from PSD
   %-----------------------------------------------------------
   n1ymin=min(oo(250:4000,ot1y));
   n1ymax=max(oo(250:4000,ot1y));
   flagt=0;
   for n=250:round(1/Ts);
       tahd=n+round(0.05/Ts);
       if (~flagt)&((oo(tahd,ot1y)-oo(n,ot1y))>0.2);
           Bv1x_min=oo(tahd,bxin);
           Vdet1y_min=oo(tahd,ot1y);
           break;
```

```
    end;
end
for n=round(1.5/Ts):4000;
    tahd=n+round(0.05/Ts);
    if (oo(tahd,ot1y)-oo(n,ot1y)<0.2);
        Bv1x_max=oo(n,bxin);
        Vdet1y_max=oo(n,ot1y);
        break;
    end
end


%-------------------------------------------------------
% ramp DFSM in theta y to get OT limits
%-------------------------------------------------------
b_test_x=const;    b_test_y=tstng;
%
set_switches
+tg
pause(2.5);
-tg
clc
clear tt oo
tt=tg.Time;
oo=tg.Output;
set(figure, 'Name', 'Baker Ramp Y, OT1', 'NumberTitle', 'off');
plot(tt,oo(:,ot1x),tt,oo(:,ot1y)),grid,zoom
xlabel('time')
ylabel('amplitude, V')
legend('OT1 X','OT1 Y')
title('Baker ramp response,theta Y, OT1')
saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Baker ramp theta Y, OT1.fig')
%-------------------------------------------------------
%   Calculate min and max values from PSD
%-------------------------------------------------------
n1xmin=min(oo(250:4000,ot1x));
n1xmax=max(oo(250:4000,ot1x));
for n=250:round(1/Ts);
    tahd=n+round(0.05/Ts);
    if (~flagt)&((oo(tahd,ot1x)-oo(n,ot1x))>0.2);
        Bv1y_min=oo(tahd,byin);
        Vdet1x_min=oo(tahd,ot1x);
        break;
    end;
end
for n=round(1.5/Ts):4000;
    tahd=n+round(0.05/Ts);
    if (oo(tahd,ot1x)-oo(n,ot1x)<0.2);
        Bv1y_max=oo(n,byin);
        Vdet1x_max=oo(n,ot1x);
        break;
    end
end
%-----------------------------------------------
% ramp RFSM in theta x to get OT limits
%-----------------------------------------------
%   set switches for RFSM test
b_test_x=const;    b_test_y=const;
n_test_x=tstng;    n_test_y=const;
%
set_switches
+tg
pause(2.5);
-tg
clc
clear tt oo
tt=tg.Time;
oo=tg.Output;
```

```
set(figure, 'Name', 'Newport Ramp response', 'NumberTitle', 'off');
plot(tt,oo(:,ot2x),tt,oo(:,ot2y),tt,oo(:,ot3x),tt,oo(:,ot3y)),grid,zoom
xlabel('time')
ylabel('amplitude, V')
legend('OT2 X','OT2 Y','OT3 X','OT3 Y')
title('Newport ramp response,theta X, OT2 and OT3')
saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Newport ramp response,theta X, OT2 and
OT3.fig')
%-------------------------------------------------------
%   Calculate min and max values from PSD
%-------------------------------------------------------
n3ymax=max(oo(250:4000,ot3y));
n3ymin=min(oo(250:4000,ot3y));
n2ymax=max(oo(250:4000,ot2y));
n2ymin=min(oo(250:4000,ot2y));
for n=250:round(1/Ts);
    tahd=n+round(0.05/Ts);
    if (~flagt)&((oo(tahd,ot3y)-oo(n,ot3y))>0.2);
        Nv3x_min=oo(tahd,nxin);
        Vdet3y_min=oo(tahd,ot3y);
        break;
    end;
end
for n=round(1.5/Ts):4000;
    tahd=n+round(0.05/Ts);
    if (oo(tahd,ot3y)-oo(n,ot3y)<0.2);
        Nv3x_max=oo(n,nxin);
        Vdet3y_max=oo(n,ot3y);
        break;
    end
end
%-----------------------------------------------
% ramp RFSM in theta y to get OT limits
%-----------------------------------------------
n_test_x=const;    n_test_y=tstng;
%
set_switches
+tg
pause(2.5);
-tg
clc
clear tt oo
tt=tg.Time;
oo=tg.Output;
set(figure, 'Name', 'Newport Ramp Y, OT2,OT3', 'NumberTitle', 'off');
plot(tt,oo(:,ot2x),tt,oo(:,ot2y),tt,oo(:,ot3x),tt,oo(:,ot3y)),grid,zoom
xlabel('time')
ylabel('amplitude, V')
legend('OT2 X','OT2 Y','OT3 X','OT3 Y')
title('Newport ramp response,theta Y, OT2 and OT3')
saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Newport ramp response,theta Y, OT2 and
OT3.fig')
%-------------------------------------------------------
%   Calculate min and max values from PSD
%-------------------------------------------------------
n3xmax=max(oo(250:4000,ot3x));
n3xmin=min(oo(250:4000,ot3x));
n2xmax=max(oo(250:4000,ot2x));
n2xmin=min(oo(250:4000,ot2x));
for n=250:round(1/Ts);
    tahd=n+round(0.05/Ts);
    if (~flagt)&((oo(tahd,ot3x)-oo(n,ot3x))>0.2);
        Nv3y_min=oo(tahd,nyin);
        Vdet3x_min=oo(tahd,ot3x);
        break;
    end;
end
for n=round(1.5/Ts):4000;
```

```
    tahd=n+round(0.05/Ts);
    if (oo(tahd,ot3x)-oo(n,ot3x)<0.2);
        Nv3y_max=oo(n,nyin);
        Vdet3x_max=oo(n,ot3x);
        break;
    end
end
%-----------------------------------------------------------
% calibration calcs mm/V on OT1/OT2
%-----------------------------------------------------------
ot1xcal=10/(n1xmax-n1xmin);   ot1ycal=10/(n1ymax-n1ymin);   %mm/V
ot2xcal=10/(n2xmax-n2xmin);   ot2ycal=10/(n2ymax-n2ymin);
ot3xcal=10/(n3xmax-n3xmin);   ot3ycal=10/(n3ymax-n3ymin);

NGmy=(Vdet3x_max-Vdet3x_min)*ot3xcal/d2/2*1000/(Nv3y_max-Nv3y_min);  % mrad/V, Newport
Y axis
NGmx=(Vdet3y_max-Vdet3y_min)*ot3ycal/d2/2*1000/(Nv3x_max-Nv3x_min);  % mrad/V, Newport
X axis
BGmx=(Vdet1y_max-Vdet1y_min)*ot1ycal/d1/2*1000/(Bv1x_max-Bv1x_min);  % mrad/V, Baker X
axis
BGmy=(Vdet1x_max-Vdet1x_min)*ot1xcal/d1/2*1000/(Bv1y_max-Bv1y_min);  % mrad/V, Baker Y
axis
otcal=1;
%-------------------------------------------------------------------
% calibration calcs mm/V on OT1/OT2 if ramp does not work
%-------------------------------------------------------------------
%  calc is voltage from max limits on OT.
%  for PSM 10-2 with OT 301 it is +/- 10 V or 20V total
%    across detector.  Use if edge voltage of detector in doubt
%   clear flag otcal if desired to use these calcs
%
if ~otcal;
    calc=20;
    ot1xcal=10/(calc);ot1ycal=10/(calc);
    ot2xcal=10/(calc);ot2ycal=10/(calc);
    ot3xcal=10/(calc);ot3ycal=10/(calc);
end


%-------------------------------------------------------------------
% step DFSM mirror in theta x
%-------------------------------------------------------------------
rampstep=step;b_test_x=tstng;
%
set_switches
+tg
pause(1.5);
-tg
clc
clear tt oo
tt=tg.Time;
oo=tg.Output;
%-------------------------------------------------------------------
% determine gain, V/mm
% assume small angle approximation
%-------------------------------------------------------------------
init1=mean(oo(10:400,ot1y));
final1=mean(oo(2000:2500,ot1y));
stepsizey1=final1-init1;
bak_x1=finval/(stepsizey1*ot1ycal);
bak_x=bak_x1;
%
init2=mean(oo(10:400,ot2y));
final2=mean(oo(2000:2500,ot2y));
stepsizey2=final2-init2;
bak_x2=finval/(stepsizey2*ot2ycal);
%
init3=mean(oo(10:400,ot3y));
```

```
final3=mean(oo(2000:2500,ot3y));
stepsizey3=final3-init3;
bak_x3=finval/(stepsizey3*ot3ycal);
%
set(figure, 'Name', 'Baker step X', 'NumberTitle', 'off');
plot(tt,oo(:,ot1x),tt,oo(:,ot1y),tt,oo(:,ot2x),tt,oo(:,ot2y),tt,oo(:,ot3x),tt,oo(:,ot3
y)),grid,zoom
xlabel('time')
ylabel('amplitude, V')
legend('OT1 X','OT1 Y','OT2 X','OT2 Y','OT3 X','OT3 Y')
title('Baker step response,theta X, OT1,OT2 and OT3')
saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Baker step response,theta X, OT1,OT2
and OT3.fig')
%-----------------------------------------------------------------
% step DFSM mirror in theta y
%-----------------------------------------------------------------
%
b_test_x=const;  b_test_y=tstng;
finval=1;
set_switches
+tg
pause(1.5);
-tg
clc
clear tt oo
tt=tg.Time;
oo=tg.Output;
init1=mean(oo(10:400,ot1x));
final1=mean(oo(2000:2500,ot1x));
stepsizex1=final1-init1;
bak_y1=finval/(stepsizex1*ot1xcal);
bak_y=bak_y1;
%
init2=mean(oo(10:400,ot2x));
final2=mean(oo(2000:2500,ot2x));
stepsizex2=final2-init2;
bak_y2=finval/(stepsizex2*ot2xcal);
%
init3=mean(oo(10:400,ot3x));
final3=mean(oo(2000:2500,ot3x));
stepsizex3=final3-init3;
bak_y3=finval/(stepsizex3*ot3xcal);
%
set(figure, 'Name', 'Baker Step Y', 'NumberTitle', 'off');
plot(tt,oo(:,ot1x),tt,oo(:,ot1y),tt,oo(:,ot2x),tt,oo(:,ot2y),tt,oo(:,ot3x),tt,oo(:,ot3
y)),grid,zoom
xlabel('time')
ylabel('amplitude, V')
legend('OT1 X','OT1 Y','OT2 X','OT2 Y','OT3 X','OT3 Y')
title('Baker step response,theta Y, OT1,OT2 and OT3')
saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Baker step response,theta X, OT1,OT2
and OT3.fig')
%-----------------------------------------------------------------
% find DFSM zero on OT1
%-----------------------------------------------------------------
%
valx_sig=getsignalid(tg, 'OT output Scopes/On Trac/MM-32 /p1');
valy_sig=getsignalid(tg, 'OT output Scopes/On Trac/MM-32 /p2');
b_test_x=const;  b_test_y=const;
set_switches
clear valx valy
ycnt=0;xcnt=0;
+tg
for i=1:5
    valx(i)=get(tg,valx_sig);
end
avgx=mean(valx);
while abs(avgx)>tol
```

```
    b_y_zero=b_y_zero-(avgx/abs(stepsizex1));
    set(tg,b_y_zero_par,b_y_zero);
    for i=1:5
        valx(i)=get(tg,valx_sig);
    end
    avgx=mean(valx);
    xcnt=xcnt+1;
    if xcnt>10
        break
    end
     end
for i=1:5
    valy(i)=get(tg,valy_sig);
end
     avgy=mean(valy);
while abs(avgy)>tol
    b_x_zero=b_x_zero-(avgy/abs(stepsizey1));
    set(tg,b_x_zero_par,b_x_zero);
    for i=1:5
        valy(i)=get(tg,valy_sig);
    end
    avgy=mean(valy);
    ycnt=ycnt+1;
    if ycnt>10
        break
    end
end
%-------------------------------------------------------------
% check b_y_zero again
%-------------------------------------------------------------
for i=1:5
    valx(i)=get(tg,valx_sig);
end
avgx=mean(valx);
xcnt=0;
while abs(avgx)>tol
    b_y_zero=b_y_zero-(avgx/abs(stepsizex1));
    set(tg,b_y_zero_par,b_y_zero);
    for i=1:5
        valx(i)=get(tg,valx_sig);
    end
    avgx=mean(valx);
    xcnt=xcnt+1;
    if xcnt>10
        break
    end
end
-tg
clc
clear tt oo
tt=tg.Time;
oo=tg.Output;
set(figure, 'Name', 'Baker zero OT1', 'NumberTitle', 'off');
plot(tt,oo(:,ot1x),tt,oo(:,ot1y)),grid,zoom
axis([-inf,inf,-1,1])
xlabel('time')
ylabel('amplitude, V')
legend('OT1 X','OT1 Y')
title('Baker zero, OT1')
saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Baker zero.fig')
b_y_zero_ot1=b_y_zero;      b_x_zero_ot1=b_x_zero;
%-------------------------------------------------------------
% step RFSM mirror in theta x
%-------------------------------------------------------------
n_test_x=tstng;   n_test_y=const;
rampstep=step;    finval=.5;
x_test=ramp;      y_test=ramp;
```

```
%
set_switches
+tg
pause(1.5);
-tg
clc
clear tt oo
tt=tg.Time;
oo=tg.Output;
init3=mean(oo(10:400,ot3y));
final3=mean(oo(1000:2500,ot3y));
init2=mean(oo(10:400,ot2y));
final2=mean(oo(1000:2500,ot2y));
stepsizey3=final3-init3;
stepsizey2=final2-init2;
%
newx3c=finval/(stepsizey3*ot3ycal);newx2c=finval/(stepsizey2*ot2ycal);
%
set(figure, 'Name', 'Newport Step X', 'NumberTitle', 'off');
plot(tt,oo(:,ot2x),tt,oo(:,ot2y),tt,oo(:,ot3x),tt,oo(:,ot3y)),grid,zoom
xlabel('time')
ylabel('amplitude, V')
legend('OT2 X','OT2 Y','OT3 X','OT3 Y')
title('Newport 0.5V step response,theta X, OT2 and OT3')
saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Newport 0.5V step response,theta X,
OT2 and OT3.fig')
%-------------------------------------------------------------
% step RFSM mirror in theta y
%-------------------------------------------------------------
n_test_x=const;   n_test_y=tstng;
finval=.5;
%
set_switches
+tg
pause(1.5);
-tg
clear tt oo
tt=tg.Time;
oo=tg.Output;
init3=mean(oo(10:400,ot3x));
final3=mean(oo(1000:2500,ot3x));
init2=mean(oo(10:400,ot2x));
final2=mean(oo(1000:2500,ot2x));
stepsizex3=final3-init3;
stepsizex2=final2-init2;
%
newy3c=finval/(stepsizex3*ot3xcal);newy2c=finval/(stepsizex2*ot2xcal);
%
set(figure, 'Name', 'Newport Step Y', 'NumberTitle', 'off');
plot(tt,oo(:,ot2x),tt,oo(:,ot2y),tt,oo(:,ot3x),tt,oo(:,ot3y)),grid,zoom
xlabel('time')
ylabel('amplitude, V')
legend('OT2 X','OT2 Y','OT3 X','OT3 Y')
title('Newport 0.5V step response,theta Y, OT2 and OT3')
saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Newport 0.5V step response,theta Y,
OT2 and OT3.fig')
%-------------------------------------------------------------
% find RFSM zero on OT3
%-------------------------------------------------------------
b_test_y=const;   b_test_x=const;
n_test_y=const;   n_test_x=const;
fct=0.1;        %   Value for sensitivity of zero algorithm steps
%
set_switches
valx_sig=getsignalid(tg, 'OT output Scopes/On Trac/MM-32 /p6');
valy_sig=getsignalid(tg, 'OT output Scopes/On Trac/MM-32 /p5');
clear valx valy
ycnt=0;xcnt=0;
```

```
+tg
for i=1:5
    valx(i)=get(tg,valx_sig);
end
avgx=mean(valx);
while abs(avgx)>tol
    n_y_zero=n_y_zero-fct*(avgx/abs(stepsizex3)*finval);
    set(tg,n_y_zero_par,n_y_zero);
    for i=1:5
        valx(i)=get(tg,valx_sig);
    end
    avgx=mean(valx);
    xcnt=xcnt+1;
    if xcnt>10
        break
    end
end
for i=1:5
    valy(i)=get(tg,valy_sig);
end
avgy=mean(valy);
while abs(avgy)>tol
    n_x_zero=n_x_zero-fct*(avgy/abs(stepsizey3)*finval);
    set(tg,n_x_zero_par,n_x_zero);
    for i=1:5
        valy(i)=get(tg,valy_sig);
    end
    avgy=mean(valy);
    ycnt=ycnt+1;
    if ycnt>10
        break
    end
end
%-------------------------------------------------------------
%check n_y_zero again
%-------------------------------------------------------------
for i=1:5
    valx(i)=get(tg,valx_sig);
end
avgx=mean(valx);
xcnt=0;
while abs(avgx)>tol
    n_y_zero=n_y_zero-fct*(avgx/abs(stepsizex3)*finval);
    set(tg,n_y_zero_par,n_y_zero);
    for i=1:5
        valx(i)=get(tg,valx_sig);
    end
    avgx=mean(valx);
    xcnt=xcnt+1;
    if xcnt>10
        break
    end
end
-tg
clear tt oo
tt=tg.Time;
oo=tg.Output;
set(figure, 'Name', 'Newport Zero OT3', 'NumberTitle', 'off');
plot(tt,oo(:,ot3x),tt,oo(:,ot3y)),grid,zoom
axis([-inf,inf,-1,1])
xlabel('time')
ylabel('amplitude, V')
legend('OT3 X','OT3 Y')
title('Newport zero, OT3')
saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Newport zero OT3.fig')
%-------------------------------------------------------------
%   hold RFSM cal values for OT3
%-------------------------------------------------------------
```

```
n_y_zero_ot3=n_y_zero;
n_x_zero_ot3=n_x_zero;
%-------------------------------------------------------------
% find RFSM zero on OT2
%-------------------------------------------------------------
finval=0.5;
n_y_zero=0;       n_x_zero=0;
b_test_y=const;   b_test_x=const;
n_test_y=const;   n_test_x=const;

fct1=0.1;         fct2=0.05;    %   Value for sensitivity of zero algorithm steps
%
set_switches
valx_sig=getsignalid(tg, 'OT output Scopes/On Trac/MM-32 /p3');
valy_sig=getsignalid(tg, 'OT output Scopes/On Trac/MM-32 /p4');
clear valx valy
ycnt=0;xcnt=0;
+tg
for i=1:5
    valx(i)=get(tg,valx_sig);
end
avgx=mean(valx);
while abs(avgx)>tol
    n_y_zero=n_y_zero-fct1*(avgx*finval/stepsizex2);
    set(tg,n_y_zero_par,n_y_zero);
    for i=1:5
        valx(i)=get(tg,valx_sig);
    end
    avgx=mean(valx);
    xcnt=xcnt+1;
    if xcnt>30
        break
    end
end
for i=1:5
    valy(i)=get(tg,valy_sig);
end
avgy=mean(valy);
while abs(avgy)>tol
    n_x_zero=n_x_zero-fct1*(avgy*finval/stepsizey2);
    set(tg,n_x_zero_par,n_x_zero);
    for i=1:5
        valy(i)=get(tg,valy_sig);
    end
    avgy=mean(valy);
    ycnt=ycnt+1;
    if ycnt>30
        break
    end
end
%-------------------------------------------------------------
%check n_y_zero again
%-------------------------------------------------------------
for i=1:5
    valx(i)=get(tg,valx_sig);
end
avgx=mean(valx);
xcnt=0;
while abs(avgx)>tol
    n_y_zero=n_y_zero-fct2*(avgx*finval/stepsizex2);
    set(tg,n_y_zero_par,n_y_zero);
    for i=1:5
        valx(i)=get(tg,valx_sig);
    end
    avgx=mean(valx);
    xcnt=xcnt+1;
    if xcnt>30
        break
```

```
    end
end
-tg
clc
clear tt oo
tt=tg.Time;
oo=tg.Output;
set(figure, 'Name', 'Newport Zero OT2', 'NumberTitle', 'off');
plot(tt,oo(:,ot2x),tt,oo(:,ot2y)),grid,zoom
axis([-inf,inf,-1,1])
xlabel('time')
ylabel('amplitude, V')
legend('OT2 X','OT2 Y')
title('Newport zero, OT2')
saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Newport zero OT2.fig')
n_y_zero_ot2=n_y_zero;
n_x_zero_ot2=n_x_zero;
%-------------------------------------------------------------
% draw boxes
%   set box size in mm: xdist or ydist is 1/2 of a side
%-------------------------------------------------------------
xdist=1;    ydist=1;
%-------------------------------------------------------------
% draw box on OT1 using DFSM
%-------------------------------------------------------------
bak_x=bak_x1*xdist;      bak_y=bak_y1*ydist;
x_test=signl;            y_test=signl;
b_test_x=tstng;          b_test_y=tstng;
n_test_x=const;          n_test_y=const;
b_x_zero=b_x_zero_ot1;
b_y_zero=b_y_zero_ot1;
%
set_switches
+tg
pause(5.1);
-tg
clear tt oo
tt=tg.Time;
oo=tg.Output;
%-------------------------------------------------------------
%  convert volts from sensor to mm
%-------------------------------------------------------------
oo(:,ot1x)=oo(:,ot1x)*ot1xcal;  oo(:,ot1y)=oo(:,ot1y)*ot1ycal;
set(figure, 'Name', 'Baker Box OT1', 'NumberTitle', 'off');
plot(oo(:,ot1x),oo(:,ot1y)),grid,zoom
axis([-5 5 -5 5])
xlabel('distance, mm')
ylabel('distance, mm')
legend('OT1')
title('Baker calibration box, OT1')
saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Baker calibration box, OT1.fig')
%-------------------------------------------------------------
% draw box on OT2 and OT3 using RFSM
%-------------------------------------------------------------
newx1=newx2c*xdist;        newy1=newy2c*ydist;
x_test=signl;              y_test=signl;
b_test_x=const;            b_test_y=const;
n_test_x=tstng;            n_test_y=tstng;
n_x_zero=n_x_zero_ot2;     n_y_zero=n_y_zero_ot2;
%
set_switches
+tg
pause(5.1);
-tg
clear tt oo
tt=tg.Time;
oo=tg.Output;
```

```
%-------------------------------------------------------------
%  convert volts from sensor to mm
%-------------------------------------------------------------
oo(:,ot3x)=oo(:,ot3x)*ot3xcal;  oo(:,ot3y)=oo(:,ot3y)*ot3ycal;
oo(:,ot2x)=oo(:,ot2x)*ot2xcal;  oo(:,ot2y)=oo(:,ot2y)*ot2ycal;
set(figure, 'Name', 'Newport Box OT2,3', 'NumberTitle', 'off');
plot(oo(:,ot2x),oo(:,ot2y),oo(:,ot3x),oo(:,ot3y)),grid,zoom
axis([-5 5 -5 5])
xlabel('distance, mm')
ylabel('distance, mm')
legend('OT2','OT3')
title('Newport calibration box, OT2 and OT3')
saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Newport calibration box, OT2 and
OT3.fig')
%-------------------------------------------------------------
%   apply offsets for OT3 and draw OT3 box
%-------------------------------------------------------------
n_y_zero=n_y_zero_ot3;     n_x_zero=n_x_zero_ot3;
newx1=newx3c*xdist;        newy1=newy3c*ydist;
%
set_switches
+tg
pause(5);
-tg
clear tt oo
tt=tg.Time;
oo=tg.Output;
oo(:,ot3x)=oo(:,ot3x)*ot3xcal;  oo(:,ot3y)=oo(:,ot3y)*ot3ycal;
set(figure, 'Name', 'Newport Box OT3', 'NumberTitle', 'off');
plot(oo(:,ot3x),oo(:,ot3y)),grid,zoom
axis([-5 5 -5 5])
xlabel('distance, mm')
ylabel('distance, mm')
legend('OT3')
title('Newport calibration box for OT 3')
saveas(gcf,'C:\Watkins\MATLAB Files\output figs\Newport calibration box for OT 3.fig')
%-------------------------------------------------------------
% return constants for newport zero on OT2
%%-------------------------------------------------------------
n_y_zero=n_y_zero_ot2;
n_x_zero=n_x_zero_ot2;
set(tg,n_x_zero_par,n_x_zero);
set(tg,n_y_zero_par,n_y_zero);
newx1=newx2c; newy1=newy2c; n_test_x=const; n_test_y=const;
%-------------------------------------------------------------
%   save gains calculated
%%-------------------------------------------------------------
save mgains b_x_zero b_y_zero bak_x1 bak_y1 bak_x2 bak_y2 bak_x3 bak_y3 ot1xcal
ot1ycal...
ot2xcal ot2ycal ot3xcal ot3ycal newx3c newy3c newx2c newy2c n_x_zero_ot2
n_x_zero_ot2...
n_y_zero_ot2 n_y_zero_ot3 finval NGmy NGmx BGmy BGmx
close_system('calibration_control');        % close the Simulink System
-tg
%-------------------------------------------------------------
%  Clear variables if desired
%-------------------------------------------------------------
reply2 = input('Enter y to clear all variables. Press Enter to continue without
clearing    ','s');
if ~isempty(reply2)
    clear
end
%-------------------------------------------------------------
%  Display Results
%-------------------------------------------------------------
disp(['Baker offset, theta X for OT1    ',num2str(b_x_zero),'  V'])
disp(['Baker offset, theta Y for OT1    ',num2str(b_y_zero),'  V'])
disp(['Baker theta X gain for OT1       ',num2str(bak_x1),'  V/mm'])
```

```
disp(['Baker theta Y gain for OT1       ',num2str(bak_y1),'  V/mm'])
disp(['Baker theta X gain for OT2       ',num2str(bak_x2),'  V/mm'])
disp(['Baker theta Y gain for OT2       ',num2str(bak_y2),'  V/mm'])
disp(['Baker theta X gain for OT3       ',num2str(bak_x3),'  V/mm'])
disp(['Baker theta Y gain for OT3       ',num2str(bak_y3),'  V/mm'])
disp(['Baker cal const theta Y          ',num2str(BGmy),'  mRad/V'])
disp(['Baker cal const theta X          ',num2str(BGmx),'  mRad/V'])
disp(['OT1 calibration const,X axis     ',num2str(ot1xcal),'  mm/V'])
disp(['OT1 calibration const,Y axis     ',num2str(ot1ycal),'  mm/V'])
disp(['OT2 calibration const,X axis     ',num2str(ot2xcal),'  mm/V'])
disp(['OT2 calibration const,Y axis     ',num2str(ot2ycal),'  mm/V'])
disp(['OT3 calibration const,X axis     ',num2str(ot3xcal),'  mm/V'])
disp(['OT3 calibration const,Y axis     ',num2str(ot3ycal),'  mm/V'])
disp(['Newport theta X gain for OT3,    ',num2str(newx3c),'  V/mm'])
disp(['Newport theta Y gain for OT3,    ',num2str(newy3c),'  V/mm'])
disp(['Newport theta X gain for OT2,    ',num2str(newx2c),'  V/mm'])
disp(['Newport theta Y gain for OT2,    ',num2str(newy2c),'  V/mm'])
disp(['Newport cal const theta Y        ',num2str(NGmy),'  mRad/V'])
disp(['Newport cal const theta X        ',num2str(NGmx),'  mRad/V'])
cd(curr_dir);
%   Clear variables no longer needed
clear reply reply1 reply2 n_test_x_par n_test_y_par b_x_zero_par b_y_zero_par
n_x_zero_par n_y_zero_par
clear x_test_par y_test_par bak_x_par bak_y_par newx_par newy_par avgx avgy n1ymin
n1ymax
clear n1xmin n1xmax n3ymax n3ymin n2ymax n2ymin n3xmax n3xmin n2xmax n2xmin otcal calc
stepsizey1 final1 init1
clear stepsizey2 final2 init2 stepsizey3 final3 init3 stepsizex2 stepsizex3 stepsizex1
clear valx valy xcnt ycnt valx_sig valy_sig fct fct1 fct2 xdist ydist
format loose
%------------------------------------------------------------------------
%               EOF
%------------------------------------------------------------------------
```

B. SAMPLE OUTPUT OF NEWPORT MIRROR CALIBRATION

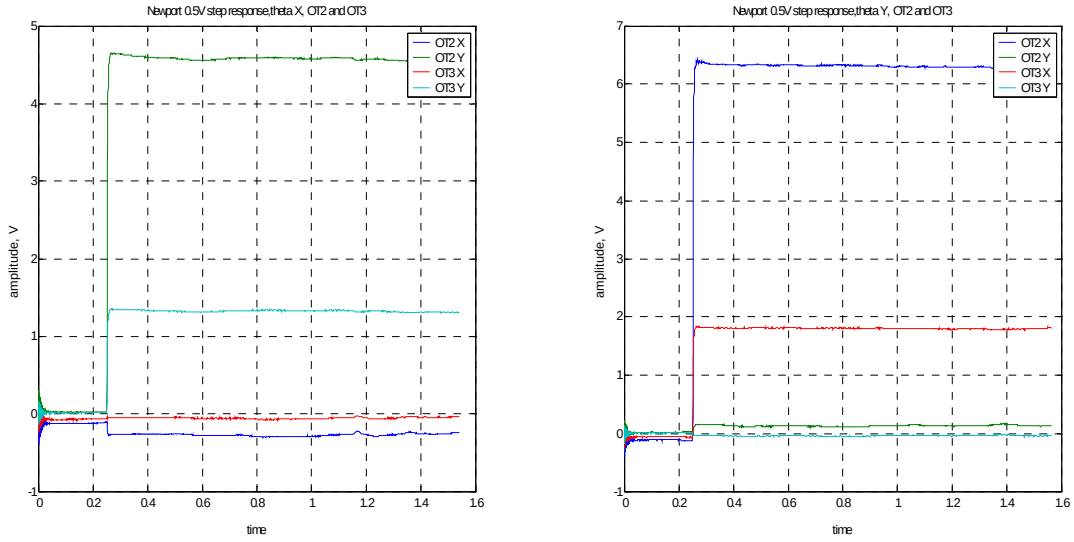


Figure 117 X and Y axis Calibration Step Response

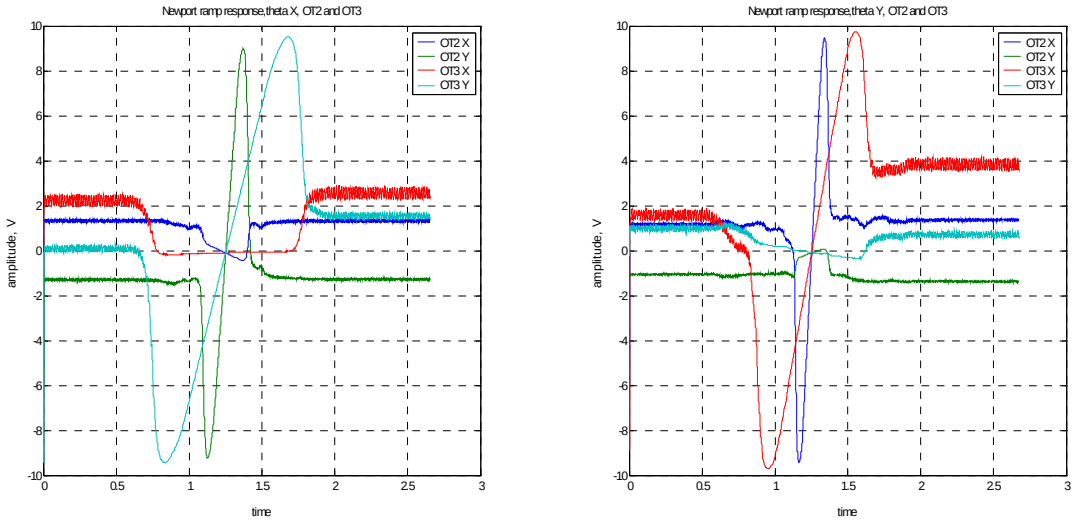


Figure 118 X and Y axis Calibration Ramp Response

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D: C-CODE FOR ADAPTIVE LATTICE ALGORITHMS

These C-coded algorithms were used in the S-function builder block from Simulink.

A. GRADIENT ADAPTIVE LATTICE

```
/* GAL algorithm from Haykin, Table 12.1 */
/*nns is the number of stages of the Lattice
  beta is a small constant, less than 0.01
  mu is the adaptation step size, a small number less than 1
inputs are u0     outputs are y0*/

int i,n,m;
{
n = *nns;
m = n + 1;


/*b0*/      y0[0] = u0[7*m];
/*f0*/      y0[5*m] = u0[7*m];

/*y0*/      y0[4*m] = u0[3*m]*u0[7*m];
/*ub0*/     y0[6*m] = *del + u0[7*m]*u0[7*m];

/*h10*/     y0[3*m] = u0[3*m] + (*mu)/y0[6*m]*u0[7*m]*u0[7*m+1];


for(i = 1;  i < m; ++i){

/*Em*/     y0[2*m+i-1] = (*beta)*u0[2*m+i-1]+(1.0-(*beta))*
                         (y0[5*m+i-1]*y0[5*m+i-1]+y0[i-1]*y0[i-1]);

/*fm*/      y0[5*m+i] = y0[5*m+i-1] + u0[m+i]*u0[i-1];

/*bm*/      y0[i] = u0[i-1] + u0[m+i]*y0[5*m+i-1];

/*km*/      y0[m+i] = u0[m+i]+(*mu)/y0[2*m+i-1]*(y0[5*m+i-1]*
                      y0[i]+u0[i-1]*y0[5*m+i]);

/*ym*/      y0[4*m+i] = y0[4*m+i-1] + y0[3*m+i]*y0[i];
/*ubm*/     y0[6*m+i] = y0[6*m+i-1] + y0[i]*y0[i];

/*hm1*/     y0[3*m+i] = u0[3*m+i] + (*mu)/y0[6*m+i]*y0[i]*u0[7*m+i];
                                  }}
```

B. GRADIENT BASED LATTICE

```
/*  Gradient Based Lattice derived from [Chen, 2001]

inputs u0,    outputs  y0
nst is the number of Lattice stages, lambda is the step size

int i,n,j;

double dele1[40],delr[40],delr1[40],ke[40],ke1[40],kr[40],
kr1[40],e[40],e1[40];

double r[40],r1[40],r2[40],ket[40],ket1[40],epsn[40],epsn1[40];
double rho[40],rho1[40],et[40],et1[40],dele[40],yref1;

n = *nst;
/*-------------------------------------------------------------
develop input vector
-------------------------------------------------------------*/
for(j = 0;  j < (n+1); ++j){
          dele[j] = u0[j];        delr[j] = u0[j+n+1];
          ke[j] = u0[j+2*(n+1)];  kr[j] = u0[j+3*(n+1)];
          e[j] = u0[j+4*(n+1)];   r[j] = u0[j+5*(n+1)];
          r1[j] = u0[j+6*(n+1)];  ket[j] = u0[j+7*(n+1)];
          et[j] = u0[j+8*(n+1)];  epsn[j] = u0[j+9*(n+1)];
          rho[j] = u0[j+10*(n+1)];     }

        for(j = 0;  j < (n+1); ++j){
          epsn1[j] = epsn[j];     rho1[j] = rho[j];
          dele1[j] = dele[j];     delr1[j] = delr[j];
          ke1[j] = ke[j];   kr1[j] = kr[j];  ket1[j] = ket[j];
          r2[j] = r1[j];    r1[j] = r[j];
          e1[j] = e[j];     et1[j] = et[j];}
          r[0] = yhat[0];  e[0] = r[0];
          et[0] = xhat[0];
          yref1 = u0[11*(n+1)];
/*-------------------------------------------------------------
   Algorithm Calculations
-------------------------------------------------------------*/
        for (i=1; i<(n+1); ++i)  {

          dele[i]=*lambda*dele1[i]+r1[i-1]*r1[i-1];
          delr[i]=*lambda*delr1[i]+e[i-1]*e[i-1];

          ke[i]=ke1[i]+1/dele1[i]*r2[i-1]*e1[i];
          kr[i]=kr1[i]+1/delr1[i]*e1[i-1]*r1[i];

          e[i]=e[i-1]-r1[i-1]*ke[i];
          r[i]=r1[i-1]-e[i-1]*kr[i];

          ket[i]=ket1[i]+1/dele[i]*r1[i-1]*et1[i];
          et[i]= et[i-1]-r[i-1]*ket[i];   }

          epsn[n] = 0;  rho[n] = 0;
```

```
        for (j=n-1;  j>0 ; --j)  {
          epsn[j] = epsn[j+1]-(rho[j+1]+yref[0]*ket[j+1])*kr[j];
          rho[j] = rho1[j+1]-epsn1[j+1]*ke[j]+yref1*ket[j+1];}

          ux[0] = -1*(epsn[1]+rho[1]+yref[0]*ket[1]);

/*-------------------------------------------------------------*/
/*  develop output vector
---------------------------------------------------------------*/
        for(j = 0;  j < (n+1); ++j){
          y0[j] = dele[j];  y0[j+n+1] = delr[j];
          y0[j+2*(n+1)] = ke[j];  y0[j+3*(n+1)] = kr[j];
          y0[j+4*(n+1)] = e[j];   y0[j+5*(n+1)] = r[j];
          y0[j+6*(n+1)] = r1[j];  y0[j+7*(n+1)] = ket[j];
          y0[j+8*(n+1)] = et[j];  y0[j+9*(n+1)] = epsn[j];
          y0[j+10*(n+1)] = rho[j];     }
          y0[11*(n+1)] = yref[0];
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX E: CALCULATION OF KALMAN AND LQR PARAMETERS

```
%   this program constructs SS rep of 2 axis, 2 input
%   control mirror system
%   States:  Vpy, Theta x, Theta-dot x
%            Vpx, Theta y, theta-dot y

%   Gm,Gmd - Mirror Sensor gain, rad/V
%   z,zd - damping;  fn,fnd - natural freq, Hz
%   wn,wnd - natural freq, rad/s
%   T,Td - Time delay for mirror/detector response, sec
%-----------------------------------------------------------------
%   Control mirror fundamentals
%-----------------------------------------------------------------
T2 = 1/15e3;    % time constant for PSD/input/output system
%-----------------------------------------------------------------
%   'natural frequency' of mirrors and mirror damping
%-----------------------------------------------------------------
fn2x = 900;  fn2y = 827;  z2 = 0.9;
wn2x=fn2x*2*pi;     wn2y=fn2y*2*pi; % convert to rads

Gm=(52.4e-3)/20;    % mirror voltage gain, rad/V
Gp=20/10e-3;        % Position Sensor Gain, V/m.
Dm=1245;            % Distance from control mirror to OT3, mm.
Dd=0.681;           % Distance from disturbance mirror to OT3, m.
Dt=1.5;             % Distance from control to OT2, m.
Ts=0.0005;          % Sample time, sec
%-----------------------------------------------------------------
%   Cross-coupling terms
%-----------------------------------------------------------------
        Ax1=-1e-2*wn2x^2;
        Ax2=-1e-2*2*wn2x*z2;
        Ay1=-1e-2*wn2y^2;
        Ay2=-1e-2*2*wn2y*z2;

%-----------------------------------------------------------------
%   SS Matrices for FSM
%-----------------------------------------------------------------
A=[ 0            1                0          0;
   -wn2x^2     -2*wn2x*z2          Ax1        Ax2;
    0            0                0          1;
   Ay1           Ay2          -wn2y^2   -2*wn2y*z2];
%-----------------------------------------------------------------
B=[0            0;
   Gm*wn2x^2    0;
   0            0;
   0        Gm*wn2y^2];
%-----------------------------------------------------------------
C=zeros(2,4);  C(1,1)=2*Dm;  C(2,3)=-C(1,1);
D=zeros(2,4);
%-----------------------------------------------------------------
```

```
    %    Noise Matrices for Kalman Filter to estimate FSM velocity
    %-----------------------------------------------------------------
     Qn = 1e-3*ones(2,2);  Nn = 1e-3*ones(2,2);  Rn = 2e-1*eye(2,2);
     Gn = 1e-2*ones(4,2);  Gn(2,1) =1e-1;  Gn(4,2)=Gn(2,1);

    %-----------------------------------------------------------------
    %    Calculate Kalman Gains
    %-----------------------------------------------------------------
     L = LQE(A,Gn,C,Qn,Rn,Nn)
    %-----------------------------------------------------------------
    %    Calculate Kalman system and convert to discrete
    %-----------------------------------------------------------------
     Ak = (A-L*C);
     Bk = [B L];
     Ck = eye(4,4);
     Dk = zeros(4,4);
     kalsysc = ss(Ak,Bk,Ck,Dk);
     kalsysd = c2d(kalsysc,Ts);
     ak = kalsysd.a;    bk = kalsysd.b;
     ck = kalsysd.c;    dk = kalsysd.d;
  %-----------------------------------------------------------------
  %values for geometry considerations of mirrors
     gty = 0.405;     gtx = 0.535;
  %-----------------------------------------------------------------
  %   State Space Model for Mirror-PSD system
  %-----------------------------------------------------------------
Asys=
[ -1/T2   gty*2*Dm/ot2ycal/T2  0      0       0                0;
   0             0             1     0       0                0;
   0        -wn2x^2     -2*wn2x*z2   0      Ax1              Ax2;
   0             0             0   -1/T2 gtx*2*Dm/ot2xcal/T2   0;
   0             0             0     0        0                1;
   0            Ay1           Ay2    0     -wn2y^2         -2*wn2y*z2];
  %-----------------------------------------------------------------
   Bsys=[0             0;
         0             0;
         Gm*wn2x^2     0;
         0             0;
         0             0;
         0             Gm*wn2y^2];

    Gsys = zeros(6,2);  Gsys(1,1)=1;  Gsys(4,2)=1;
  %-----------------------------------------------------------------
  %    Convert output to mm, ot2xcal/ycal are from calibration program
  %-----------------------------------------------------------------
    Csys = eye(6,6);    Dsys = 0;  Csys(4,4)=-1;
    Csys2 = zeros(2,6);  Csys2(1,1) = ot2ycal;  Csys2(2,4)=-ot2xcal;
  %-----------------------------------------------------------------
  %    Modify SS system to include integrator (after Ogata)
  %-----------------------------------------------------------------
    Cint = zeros(2,6);  Cint(1,1) = ot2xcal;  Cint(2,4) = ot2ycal;
    Aint = [Asys zeros(6,2);-Cint zeros(2,2)]
    Bint = [Bsys;zeros(2,2)]
  %-----------------------------------------------------------------
```

```
%   Weighting Matrices for LQRD function
%-------------------------------------------------------------
    Qa=1000;
    Q=eye(8,8);  Q(1,1)=Qa;   Q(4,4)=Qa;
    R=0.1*eye(2,2);

    K = LQRD(Aint,Bint,Q,R,Ts);
%-------------------------------------------------------------
%   Convert SS continuous system to discrete
%-------------------------------------------------------------
    dasysc2 = ss(Asys,Bsys,Csys2,zeros(2,2));
    dasysd2 = c2d(dasysc2,Ts);
    am2=dasysd2.a;
    bm2=dasysd2.b;
    cm2=dasysd2.c;
    dm2=dasysd2.d;
%-------------------------------------------------------------
%   EOF
%-------------------------------------------------------------
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX F: X AXIS RESULTS FOR MULTIPLE FREQUENCY PLUS NOISE CASES

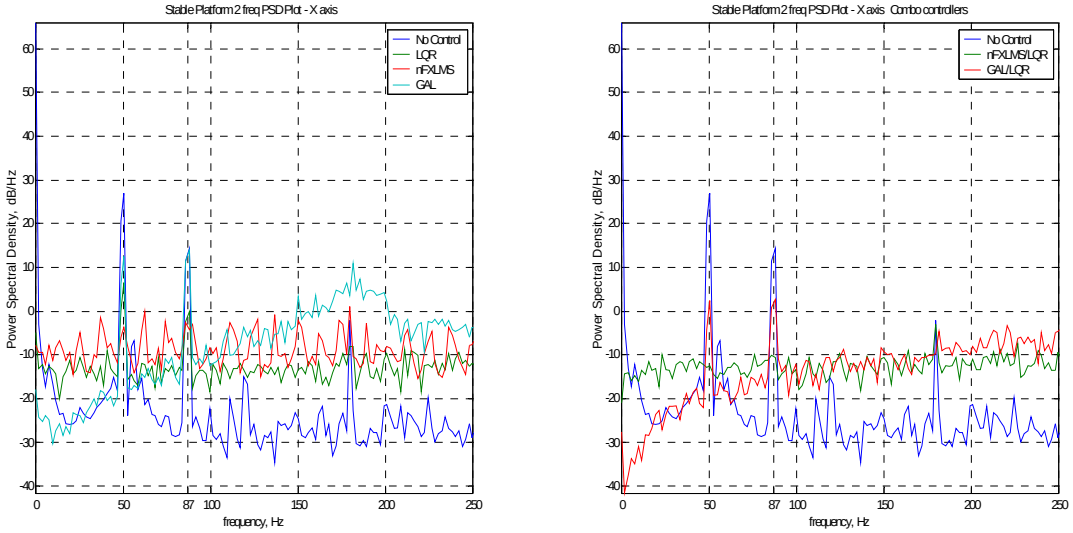


Figure 119 PSD of X Axis Response for Stationary Platform, 2 Freq. Disturbance

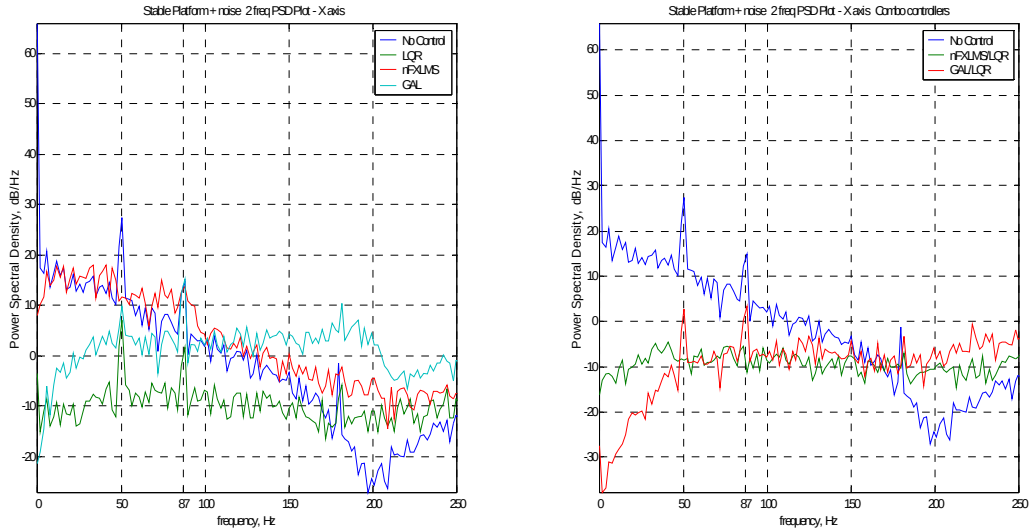


Figure 120 PSD of X axis Response for Stationary Platform 2 Freq. plus noise Disturbance

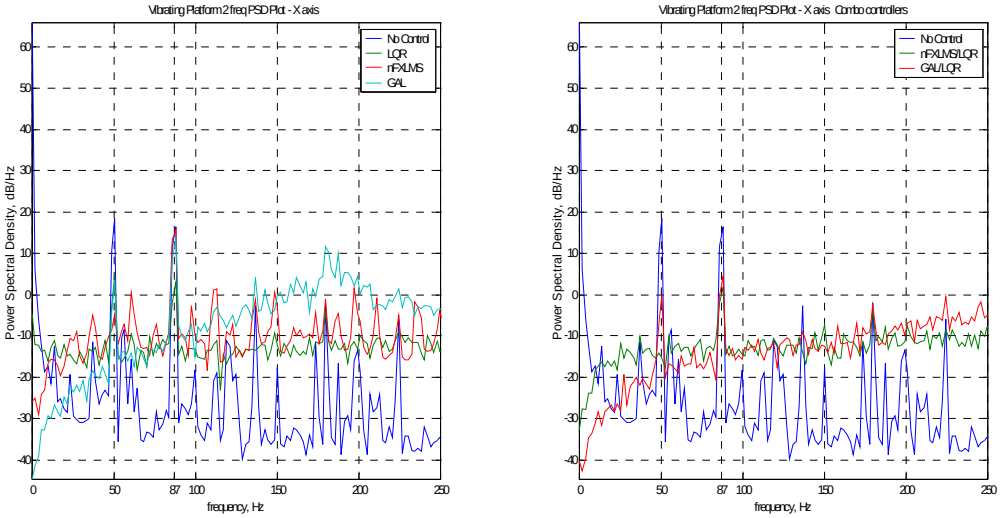


Figure 121 PSD of X Axis Response for Vibrating Platform, 2 Freq. Disturbance

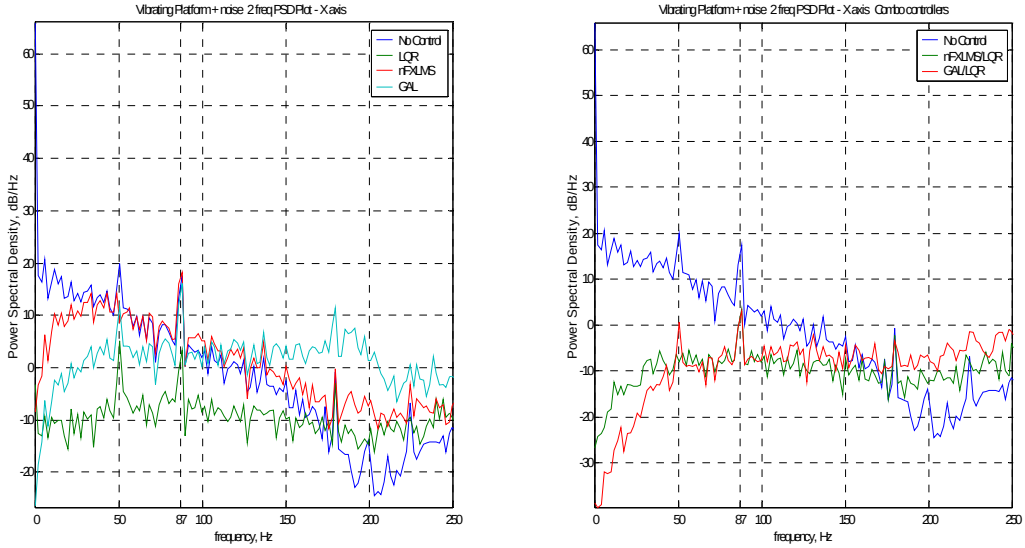


Figure 122 PSD of X axis Response for Vibrating Platform 2 Freq. plus noise Disturbance

LIST OF REFERENCES

[Astrom, 1995] Astrom, K.J., and Wittenmark, B., *Adaptive Control*, 2nd Ed. pp. 2-3, Addison-Wesley, Menlo Park, CA, 1995

[Boelitz, 2003] Boelitz, F.W., Gibson, J.S., Walter, R.E. and Washburn, D.C., "Active Vibration Cancellation for ABLE ACE," Unpublished.

[Burdess, 1985] Burdess, J.S. and Metcalfe, A.V., "The Active Control of Forced Vibration Produced by Arbitrary Disturbances," *Journal of Vibration, Acoustics, Stress, and Reliability in Design*, Vol. 107, pp. 33-37, 1985

[Burg, 1968] Burg, J.P., "A New Analysis Technique for time Series Data," *NATO Advanced Study Institute on Signal Processing*, Enschede, The Netherlands.

[Burgess, 1981] Burgess, J.C., "Active adaptive sound control in a duct: A computer simulation," *Journal of the Acoustical Society of America*, Vol. 70, pp. 715-726, September 1981

[Chen, 2001] Chen, S.-J. and Gibson, J.S., "Feedforward Adaptive Noise Control with Multivariable Gradient Lattice Filters", *IEEE Transactions on Signal Processing*, Vol. 49, No. 3, pp 511-519, 2001

[Elliott, 1993] Elliott, S. J., and Nelson, P.A., "Multichannel active sound control using adaptive filtering," in *Proceedings of the ICASSP*, 1988, pp. 2590-2593

[Etter, 1981] Etter, D.M. and Stearns, S.D., "Adaptive estimation of time delays in sampled data systems," *IEEE Transactions in Acoustic, Speech and Signal Processing*, vol. ASSP-29, pp. 582-587, June 1981.

[Glaese, 2000] Glaese, R.M., Anderson, E.H., and Janzen, P.C., "Active Suppression of Acoustically Induced Jitter for the Airborne Laser," *Proceedings of the SPIE Conference on Laser Weapons Technology*, SPIE paper 4034-19, pp. 157-162, April 2000.

[Griffiths, 1978] Griffiths, L.J., "An adaptive lattice structure for noise-canceling applications," *Proceedings of the IEEE ICASSP*, pp. 87-90, 1978

[Haykin, 2002] Haykin, S. *Adaptive Filter Theory*, 4th Ed., Prentice-Hall, Upper Saddle River, NJ, 2002

[Hunnicutt, 2003] Hunnicutt, D., manager On-Trak Corporation, phone conversation of April 11, 2003.

[Jiang, 1995] Jiang, S-B., and Gibson, J.S., "An Unwindowed Multichannel Lattice Filter with Orthogonal Channels," *IEEE Transactions on Signal Processing*, Vol. 43, No. 12, pp. 2831, December 1995

[Kuo, 1996] Kuo Sen M., and Morgan, Dennis R., *Active Noise Control Systems Algorithms and DSP Implementations,* John Wiley & Sons, New York, 1996.

[Kuo, 1998a] Kuo, Sen M., Kong, Xuan, Chen, Shaojie, and Hao, Wenge, "Analysis and Design of Narrowband Active Noise Control Systems," *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1998*, Volume: 6, 12-15 May 1998, Pages:3557 – 3560.

[Kuo, 1998b] Kuo, Sen, M., and Chung, K.M., "Narrowband Active Noise Control Using Adaptive Delay Filter," *IEEE Signal Processing Letters*, Vol. 5, No. 12, pp. 309-311, December, 1998

[Lueg, 1937] Lueg, P., *Verfahren zur Dampfung von Schallschwingungen*, German Patent, DRP No. 655,508, Filed January 1933, Issued December 1937

[Makhoul, 1978] Makhoul, J., "A class of all-zero lattice digital filters: Properties and applications," *IEEE Transactions on Acoustic Speech Signal Processing*, Vol. ASSP-26, pp. 304-314, 1978

[McEver, 2001] McEver, M.A., and Clark, R.L., "Active Jitter Suppression of Optical Structures," *Proceedings of the SPIE* Vol. 4327, pp. 596-598, 2001

[Newport, 2002] Newport Instruction Manual (Draft) for the FSM200 Fast Steering Mirror and FSM-CD100 Controller/Driver, Revision N, October 18, 2002

[Ogata, 2002] Ogata, K., *Modern Control Engineering*, 4th Ed., Prentice-Hall, Upper Saddle River, NJ, 2002

[Schafer, 1985] Schafer, B.E. and Holzach, H., "Experimental Research on Flexible Beam Modal Control," *Journal of Guidance and Control*, Vol. 8, No. 5, pp. 597-604, 1985

[Skormin, 1995] Skormin, V.A., Tascillo, M.A., and Busch, T.E., "Adaptive jitter rejection technique applicable to airborne laser communication systems," *Optical Engineering,* Vol. 34, pp. 1267, May 1995

[Skormin, 1997a] Skormin, V.A. and Busch, T.E., "Experimental implementation of model reference control for fine tracking mirrors," *Proceedings of the SPIE* Vol. 2990, pp. 183-189, February 1997

[Skormin, 1997b] Skormin, V.A., Tascillo, M.A. and Busch, T.E., "Demonstration of a Jitter Rejection Technique for Free-Space Laser Communication" *IEEE Transactions on Aerospace and Electronic Systems,* Vol. 33, No. 2, pp. 571-574, April 1997

[Shan, 1988] Shan, T.J., and Kailath, T., "Adaptive algorithms with an automatic gain control feature," *IEEE Transactions on Circuits and Systems*, CAS-35, 122-127, January, 1988

[Takagami, 1985] Takagami, T. and Jimbo, Y., "Study of an Active Vibration Isolation System – A Learning Control Method," *Journal of Low Frequency Noise and Vibration*, Vol. 4, No. 3, pp 104-119, 1985

[Watkins , 2004] Watkins, R.J., Chen, H.J., Agrawal, B.N., Shin, Y.S., "Optical Beam Jitter Control," *Proceedings of the SPIE Conference on Lasers and Applications in Science and Technology*, SPIE paper 5338-20, January 2004.

[Widrow, 1960] Widrow, B., and Hoff, M.E., Jr., "Adaptive Switching Circuits," IREWESCON Conv. Rec., Pr. 4, 1960, pp. 96-104

[Widrow, 1970] Widrow, B., "Adaptive Filters," in *Aspects of Network and Systems Theory*, R. E. Kalman and N. DeClaris, eds., New York: Holt, Rinehart and Winston, 1970, pp. 563-587

[Widrow, 1975] Widrow, B., Glover, J.R., Jr., McCool, J.M., Kaunitz, J., Williams, C.S., Hearn, R.J.,Zeidler, J.R., Dong, E., Jr. and Goodlin, R.C., "Adaptive Noise Canceling: Principles and Applications," *Proceedings of the IEEE*, Vol. 63, No. 12, pp. 1692-1716, 1975

[Widrow, 1981] Widrow, B., Shur, D. and Shaffer, S. "On adaptive inverse control," *Proceedings of the 15th Asilomar Conference*, pp. 185-189, 1981

[Widrow, 1985] Widrow, B., *Adaptive Signal Processing*, 1st Ed., Prentice-Hall, Upper Saddle River, NJ, 1985

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, VA

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, CA

3. Prof. Brij Agrawal
   Department of Mechanical and Astronautical Engineering
   Naval Postgraduate School
   Monterey CA

4. CDR Joe Watkins
   Department of Mechanical Engineering
   United States Naval Academy
   Annapolis MD

5. Prof. Young Shin
   Department of Mechanical and Astronautical Engineering
   Naval Postgraduate School
   Monterey CA

6. Prof. Anthony Healey
   Department of Mechanical and Astronautical Engineering
   Naval Postgraduate School
   Monterey CA

7. Prof. Joshua Gordis
   Department of Mechanical and Astronautical Engineering
   Naval Postgraduate School
   Monterey CA

8. Prof. Fotis Papoulias
   Department of Mechanical and Astronautical Engineering
   Naval Postgraduate School
   Monterey CA

9. Prof. Roberto Cristi
Department of Mechanical and Astronautical Engineering
Naval Postgraduate School
Monterey CA

10. Dr. Donald Washburn
Air Force Research Laboratory
Kirtland AFB
Albuquerque NM